J Data Semant (2016) 5:55–75
DOI 10.1007/s13740-015-0055-y

ORIGINAL ARTICLE

Answering Fuzzy Conjunctive Queries Over Finitely Valued Fuzzy Ontologies

Stefan Borgwardt[1] · Theofilos Mailis[2] · Rafael Peñaloza[3] · Anni-Yasmin Turhan[1]

Received: 15 January 2015 / Revised: 9 December 2015 / Accepted: 11 December 2015 / Published online: 28 January 2016

Abstract Fuzzy Description Logics (DLs) provide a means for representing vague knowledge about an application domain. In this paper, we study fuzzy extensions of conjunctive queries (CQs) over the DL $\mathcal{SROIQ}$ based on finite chains of degrees of truth. To answer such queries, we extend a well-known technique that reduces the fuzzy ontology to a classical one, and use classical DL reasoners as a black box. We improve the complexity of previous reduction techniques for finitely valued fuzzy DLs, which allows us to prove tight complexity results for answering certain kinds of fuzzy CQs. We conclude with an experimental evaluation of a prototype implementation, showing the feasibility of our approach.

1 Introduction

Description Logics (DLs) are a family of knowledge representation languages with unambiguous syntax and well-defined semantics that are widely used to represent the conceptual knowledge of an application domain in a structured and formally well-understood manner.

DLs have been successfully employed to formulate ontologies for a range of knowledge domains, in particular for the bio-medical sciences. Prominent examples of ontologies in these areas are the Gene Ontology,[1] and the ontology SNOMED CT.[2] Arguably the largest success of DLs to date is that they provide the formal foundation for the standard web ontology language OWL, a milestone for the Semantic Web. More precisely, the current version of the web ontology language, OWL 2, is based on the very expressive DL $\mathcal{SROIQ}$.[3]

In DLs, knowledge is represented through *concepts* that describe collections of objects (that is, correspond to unary predicates from first-order logic), and *roles* that define relations between pairs of objects (binary predicates). To encode the actual knowledge of the domain, DLs employ different kinds of *axioms*. These axioms restrict the possible interpretations of the concepts and roles.

For example, we can express the fact that cpuA is an overused CPU, and that every server that has a part that is overused is a server with limited resources through the axioms

$$(\mathsf{CPU} \sqcap \mathsf{Overused})(\mathsf{cpuA}) \tag{1}$$

$$\mathsf{Server} \sqcap \exists \mathsf{hasPart.Overused} \sqsubseteq \mathsf{ServerWithLimitedResources} \tag{2}$$

An axiom of the form (1) is called an *assertion*, while (2) is a *general concept inclusion* (GCI).

✉ Stefan Borgwardt
stefan.borgwardt@tu-dresden.de

Theofilos Mailis
theofilos@image.ntua.gr

Rafael Peñaloza
rafael.penaloza@unibz.it

Anni-Yasmin Turhan
anni-yasmin.turhan@tu-dresden.de

[1] Chair for Automata Theory, Institute for Theoretical Computer Science, Technische Universität Dresden, Dresden, Germany

[2] Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, Athens, Greece

[3] KRDB Research Centre, Free University of Bozen-Bolzano, Bolzano, Italy

[1] http://geneontology.org/.

[2] http://ihtsdo.org/snomed-ct/.

[3] http://www.w3.org/TR/owl2-overview/.

It has been widely argued that many application domains require the representation of vague concepts, for which it is impossible to precisely characterize the objects that belong to these concepts, and distinguish them from those who do not belong to them [40]. A simple example of such a concept is that of an overused CPU. While it is easy to state that a CPU that is running permanently at its maximum capacity is overused, and one that is not being used at all is not overused, there is no precise usage point where a CPU starts (or stops) being overused. Fuzzy Description Logics have been proposed to alleviate this problem. In these logics, objects are assigned a *membership degree*, typically a number between 0 and 1, expressing "how much" they belong to a given concept.

In general, the higher the degree of an object, the more it belongs to the concept. To represent vague knowledge, axioms are also extended to restrict the possible degrees that the interpretations may use.

Thus, one can express that cpuA is overused with degree at least 0.8 through the assertion $\mathsf{Overused}(\mathsf{cpuA}) \geqslant 0.8$.

Formally, fuzzy DLs generalize classical DLs by interpreting concepts and roles as fuzzy unary predicates and fuzzy binary predicates, respectively. Hence, fuzzy DLs can be seen as sublogics of fuzzy first-order logic. Adopting this view, one can use a triangular norm (*t-norm*) and its associated operators to interpret the different logical constructors. Each t-norm then defines a specific family of fuzzy DLs.

It has been shown that reasoning in fuzzy DLs easily becomes undecidable, if infinitely many membership degrees are allowed [1,12]. In fact, these undecidability results hold even for relatively inexpressive fuzzy DLs. This has motivated the study of *finitely valued* fuzzy DLs.

It is known that the complexity of standard reasoning tasks in expressive DLs is not affected by the use of t-norm-based finitely valued semantics [13,14]. Unfortunately, the automata-based techniques exploited in [13] cannot be easily adapted to obtain complexity bounds for the problem of answering conjunctive queries in these logics. Moreover, despite providing optimal complexity bounds, automata-based methods are not used in practice due to their bad best-case behavior.

A different approach for reasoning in the presence of finitely many membership degrees is *crispifying*; i.e., transforming a finitely valued ontology into an "equivalent" classical ontology, from which the relevant membership degrees can be read [2,4,11,37].

Reasoning in finitely valued fuzzy DLs is thus reduced to reasoning in classical DLs, for which very efficient methods have already been developed and implemented. The main drawback of the translation described in [4,11] is that it may introduce an exponential blow-up of the ontology, thus affecting the efficiency of the overall method.

In this paper, we adapt the crispification approach for answering conjunctive queries in expressive finitely valued fuzzy DLs. The problem of answering conjunctive queries has recently received much attention as a powerful means to access facts encoded in an ontology. For example, using a fuzzy conjunctive query it is possible to ask for all pairs of servers and CPUs such that the CPU is an overused part (to degree at least 0.6) of the server as follows:

$$\{\mathsf{Server}(x) \geqslant 1,\ \mathsf{hasPart}(x, y) \geqslant 1,\ \mathsf{CPU}(y) \geqslant 1,\\ \mathsf{Overused}(y) \geqslant 0.6\}.$$

The crispification approach allows us to effectively answer conjunctive queries over finitely valued ontologies, by reusing the methods developed for the classical case. Once the ontology is crispified, this approach calls a classical conjunctive query answering engine as a black-box procedure. Thus, any optimization developed for the classical case automatically improves the performance for the finitely valued scenario. What remains to be addressed is the exponential blow-up of the ontology, if done according to [4,11]. We strengthen our results by providing a linear preprocessing step that avoids the exponential blow-up produced by this crispification. Using this preprocessing of the finitely valued ontology, we can guarantee that the classical ontology produced is only polynomially larger than the original input. In particular, this means that the classical query answering engine becomes able to provide answers more efficiently over classical ontologies of lesser size.
The contributions of this paper are the following:

- We prove that some of the previous crispification algorithms [8,10,28] are incorrect for qualified number restrictions (indicated by the letter $\mathcal{Q}$ in the name of DLs) by means of a counter-example (see Example 8).
- We discuss a possible way to reduce such qualified number restrictions, but which depends on the presence of so-called Boolean role constructors [34] in the DL (see Sect. 3.1).
- We improve the reduction from finitely valued $\mathcal{SROIN}$ ontologies to classical $\mathcal{SROIN}$ ontologies by introducing a linear normalization step ($\mathcal{N}$ stands for *unqualified* number restrictions, see Sect. 3).
- We extend the crispification approach to answering different types of fuzzy conjunctive queries in the finitely valued setting and we prove correctness of the obtained methods (see Sect. 4). This approach works for any known crispification algorithm, in particular specialized ones that correctly reduce number restrictions.
- We assess the complexity of the presented conjunctive query answering technique for a family of fuzzy extensions of (sublogics of) $\mathcal{SROIQ}$ with regard to two types of conjunctive queries that use membership degrees.

- We provide an evaluation of a prototype implementation of our methods over the LUBM ontology benchmark [23] based on the reduction-based DeLorean reasoner [5] for fuzzy ontologies and a standard query answering reasoner for crisp ontologies (PAGOdA [42]).

A preliminary version of this paper can be found in [28], where the incorrect reduction of number restrictions was still used. We also extend here that earlier paper by full proofs for the correctness of the crispification procedure, something which has not been done before in the literature. Finally, we optimize the crispification procedure described [28] to eliminate an exponential blow-up inherent in some of the previous crispification proposals.

The rest of the paper is structured as follows: Sect. 2 introduces the syntax and semantics of finitely valued fuzzy DLs based on $\mathcal{SROIQ}$. Section 3 describes our improved reduction procedure from fuzzy to classical ontologies. Section 4 presents the actual reduction from fuzzy to classical conjunctive query answering. Section 5 provides an evaluation of a prototype implementation over (fuzzified versions of) the LUBM ontology benchmark [23]. Finally, Sect. 6 presents the current literature on reduction techniques and conjunctive query answering for fuzzy DLs and Sect. 7 summarizes the paper and mentions directions for future work.

2 Preliminaries

We first introduce a class of finite chains, together with some basic operations over them. Afterwards, we formally define the fuzzy extension of $\mathcal{SROIQ}$, whose semantics is based on these chains.

2.1 Finite Fuzzy Logics

The semantics of fuzzy DLs is based on truth structures endowed with additional operators for interpreting the logical constructors. We consider arbitrary *finite* total orders (or *chains*). Since the names of the truth degrees in a chain are not relevant, we consider in the following only the canonical chain of n elements $\mathscr{C} := \left\{0, \frac{1}{n-1}, \ldots, \frac{n-2}{n-1}, 1\right\}$, in the usual order. We denote $\mathscr{C}\backslash\{0\}$ by $\mathscr{C}_{>0}$. We use the notation d_{next} to refer to the *direct upper neighbour* of d in $\mathscr{C}$, which is the unique smallest element strictly larger than d. We now consider tuples of the form $(\mathscr{C}, \otimes, \Rightarrow, \ominus, \oplus)$ that specify the chain together with the operators used for interpreting conjunction, implication, negation, and disjunction, respectively.

The largest family of operators used for fuzzy semantics is based on *t-norms*, which are associative, commutative binary operators that are monotonic in both arguments and have identity 1. These binary operators, denoted by $\otimes$, are used in mathematical fuzzy logic to interpret conjunction. The *residuum* $\Rightarrow$ is a binary operator which is used to interpret implication. It is uniquely defined by the property that

$$(x \otimes y) \leqslant z \text{ iff } y \leqslant (x \Rightarrow z) \text{ for all } x, y, z \in \mathscr{C}.$$

The *residual negation* $\ominus$ is defined simply as $\ominus x := x \Rightarrow 0$. Finally, the *t-conorm*, used for the disjunction, is defined as $x \oplus y := 1 - ((1-x) \otimes (1-y))$ for all $x, y \in \mathscr{C}$. Two prominent families of operators are based on the *finite Gödel t-norm* and the *finite Łukasiewicz t-norm* (see Table 1). Note, however, that we do not restrict our considerations to only those logics listed in Table 1; our results are valid for any semantics based on a finite t-norm.

An alternative to t-norm-based approaches for interpreting the logical connectives in fuzzy logics is the so-called Zadeh family of operators, shown in the first row of Table 1. Intuitively, the *Zadeh family* can be seen as a combination of the Gödel and the Łukasiewicz operators.

2.2 The Fuzzy DL $\mathscr{C}$-$\mathcal{SROIQ}$

We introduce finitely valued, fuzzy extensions of the classical description logic $\mathcal{SROIQ}$ [25]—one of the most expressive decidable DLs which provides the direct model-theoretic semantics of the standardized ontology language for the Semantic Web OWL 2. It has been shown that reasoning in finitely valued fuzzy extensions of $\mathcal{SROIQ}$ can be reduced to reasoning in classical $\mathcal{SROIQ}$ [3,10,37]. This reduction technique will be considered in detail in Sect. 3.

Consider three countable and pairwise disjoint sets of *individual names* $\mathsf{N_I}$, *concept names* $\mathsf{N_C}$, and *role names* $\mathsf{N_R}$. Individual names refer to single elements of an application domain, concept names describe sets of elements, and role names binary relations between elements. Based on these, complex *concepts* and *roles* can be built using different con-

Table 1 Families of fuzzy logic operators

Name	Conjunction $x \otimes y$	Disjunction $x \oplus y$	Negation $\ominus x$	Implication $x \Rightarrow y$
Zadeh	$\min(x, y)$	$\max(x, y)$	$1 - x$	$\max(1 - x, y)$
Gödel	$\min(x, y)$	$\max(x, y)$	$\begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{if } x > 0 \end{cases}$	$\begin{cases} 1 & \text{if } x \leqslant y \\ y & \text{if } x > y \end{cases}$
Łukasiewicz	$\max(x + y - 1, 0)$	$\min(x + y, 1)$	$1 - x$	$\min(1 - x + y, 1)$

structors. More precisely, a *(complex) role* is either of the form r or r^- (*inverse role*), for $r \in \mathsf{N_R}$, or it is the *universal role* u. Similarly, *(complex) concepts* are built inductively from concept names using the following constructors:

- $\top$ (*top concept*),
- $\bot$ (*bottom concept*),
- $C \sqcap D$ (*conjunction*),
- $C \sqcup D$ (*disjunction*),
- $\neg C$ (*negation*),
- $\forall r.C$ (*value restriction*),
- $\exists r.C$ (*existential restriction*),
- $\{d_1/a_1, \ldots, d_m/a_m\}$ (*fuzzy nominal*),
- $\geq m\, r.C$ (*at-least restriction*),
- $\leq m\, r.C$ (*at-most restriction*), and
- $\exists r.\mathrm{Self}$ (*local reflexivity*),

where C, D are concepts, r is a role, m is a natural number, $a_1, \ldots, a_m \in \mathsf{N_I}$, and $d_1, \ldots, d_m \in \mathscr{C}_{>0}$.

An *ontology* $\mathcal{O}$ consists of the intensional and the extensional knowledge related to an application domain. The intensional knowledge, i.e., the general knowledge about the application domain, is expressed through

- a *TBox* $\mathcal{T}$, a set of finitely many *(fuzzy) general concept inclusion (GCIs) axioms* of the form $\langle C \sqsubseteq D \geqslant d\rangle$, where $d \in \mathscr{C}_{>0}$, and
- an *RBox* $\mathcal{R}$, a finite set of *role axioms*, which are statements of the following form:
    - $\langle r_1 \ldots r_m \sqsubseteq r \geqslant d\rangle$ (*(fuzzy) complex role inclusion*),
    - $\mathrm{trans}(r)$ (*transitivity*),
    - $\mathrm{dis}(r_1, r_2)$ (*disjointness*),
    - $\mathrm{ref}(r)$ (*reflexivity*),
    - $\mathrm{irr}(r)$ (*irreflexivity*),
    - $\mathrm{sym}(r)$ (*symmetry*), or
    - $\mathrm{asy}(r)$ (*asymmetry*),

where $r, r_1, \ldots, r_m$ are roles and again $d \in \mathscr{C}_{>0}$.

The extensional knowledge, which refers to the particular knowledge about specific facts or situations, is expressed by an *ABox* $\mathcal{A}$ containing a finite set of statements about individuals of the form:

- $\langle C(a) \bowtie d\rangle$ (*concept assertion*),
- $\langle r(a, b) \bowtie d\rangle$ (*role assertion*),
- $a \neq b$ (*individual inequality assertion*), or
- $a = b$ (*individual equality assertion*),

where $a, b \in \mathsf{N_I}$, C is a concept, r is a role, $d \in \mathscr{C}$, and $\bowtie \in \{\leqslant, \geqslant\}$. If $\bowtie$ is $\geqslant$, then we again consider only values $d > 0$; dually, for assertions using $\leqslant$ we assume that $d < 1$.

For any axiom of the form $\langle \alpha \geqslant 1\rangle$, we may simply write α. Finally, an *ontology* is a tuple $\mathcal{O} = (\mathcal{A}, \mathcal{T}, \mathcal{R})$ consisting of an ABox $\mathcal{A}$, a TBox $\mathcal{T}$, and an RBox $\mathcal{R}$.

To ensure decidability of classical $\mathcal{SROIQ}$, a set of restrictions regarding the use of roles is imposed. For example, transitive roles are not allowed to occur in number restrictions (for more details see [25]). The same restrictions are also adopted for fuzzy extensions of $\mathcal{SROIQ}$ [3,7,10]. However, they are not essential for our purposes, as all results presented in this paper hold regardless of these restrictions (except, of course, for the complexity results of Sect. 4.2).

The semantics of $\mathscr{C}$-$\mathcal{SROIQ}$ is defined via interpretations. A *(fuzzy) interpretation* is a pair $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, consisting of a non-empty set $\Delta^\mathcal{I}$ (called the *domain*) and an *interpretation function* $\cdot^\mathcal{I}$ 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 fuzzy set $A^\mathcal{I} \colon \Delta^\mathcal{I} \to \mathscr{C}$, and every role name $r \in \mathsf{N_R}$ to a fuzzy binary relation $r^\mathcal{I} \colon \Delta^\mathcal{I} \times \Delta^\mathcal{I} \to \mathscr{C}$. This function is extended to complex roles and complex concepts as described in Table 2. Note that the usual fuzzy semantics of existential and value restrictions, number restrictions, and role inclusions formally require the computation of an infimum or supremum over all domain elements. However, since $\mathscr{C}$ is a finite chain, in our case these are actually minima or maxima, respectively.

For the two-valued chain $\mathscr{C} = \{0, 1\}$, we obtain the semantics of classical $\mathcal{SROIQ}$, since then all fuzzy operators correspond to their classical counterparts. In this setting, it is more natural to treat $C^\mathcal{I}$ as a subset of $\Delta^\mathcal{I}$, given by its characteristic function $C^\mathcal{I} \colon \Delta^\mathcal{I} \to \{0, 1\}$ (and analogously

Table 2 Syntax and semantics of concepts in $\mathscr{C}$-$\mathcal{SROIQ}$

Concept C	Semantics $C^\mathcal{I}(x)$ for $x \in \Delta^\mathcal{I}$
$\top$	1
$\bot$	0
$C \sqcap D$	$C^\mathcal{I}(x) \otimes D^\mathcal{I}(x)$
$C \sqcup D$	$C^\mathcal{I}(x) \oplus D^\mathcal{I}(x)$
$\neg C$	$\ominus C^\mathcal{I}(x)$
$\forall r.C$	$\inf_{y \in \Delta^\mathcal{I}} r^\mathcal{I}(x, y) \Rightarrow C^\mathcal{I}(y)$
$\exists r.C$	$\sup_{y \in \Delta^\mathcal{I}} r^\mathcal{I}(x, y) \otimes C^\mathcal{I}(y)$
$\{d_1/o_1, \ldots, d_m/o_m\}$	$\begin{cases} d_i & \text{if } x = o_i^\mathcal{I},\ i \in \{1, \ldots, m\} \\ 0 & \text{otherwise} \end{cases}$
$\geq m\, r.C$	$\sup_{\substack{y_1,\ldots y_m \in \Delta^\mathcal{I} \\ \text{pairwise different}}} \min_{i=1}^{m} r^\mathcal{I}(x, y_i) \otimes C^\mathcal{I}(y_i)$
$\leq m\, r.C$	$\inf_{\substack{y_1,\ldots y_{m+1} \in \Delta^\mathcal{I} \\ \text{pairwise different}}} \ominus \left(\min_{i=1}^{m+1} r^\mathcal{I}(x, y_i) \otimes C^\mathcal{I}(y_i) \right)$
$\exists r.\mathrm{Self}$	$r^\mathcal{I}(x, x)$
Role s	Semantics $s^\mathcal{I}(x, y)$ for $x, y \in \Delta^\mathcal{I}$
r^-	$r^\mathcal{I}(y, x)$
u	1

Table 3 Syntax and semantics of axioms in $\mathscr{C}$-$\mathcal{SROIQ}$

ABox	Semantics
$C(a) \bowtie d$	$C^{\mathcal{I}}(a^{\mathcal{I}}) \bowtie d$
$r(a,b) \bowtie d$	$r^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}}) \bowtie d$
$a \neq b$	$a^{\mathcal{I}} \neq b^{\mathcal{I}}$
$a = b$	$a^{\mathcal{I}} = b^{\mathcal{I}}$
TBox	Semantics (for all $x \in \Delta^{\mathcal{I}}$)
$\langle C \sqsubseteq D \geqslant d \rangle$	$C^{\mathcal{I}}(x) \Rightarrow D^{\mathcal{I}}(x) \geqslant d$
RBox	Semantics (for all $x, y, z \in \Delta^{\mathcal{I}}$)
$\langle r_1 \ldots r_n \sqsubseteq r \geqslant d \rangle$	$\left(\sup_{x_1,\ldots,x_{n-1} \in \Delta^{\mathcal{I}}} r_1^{\mathcal{I}}(x, x_1) \otimes \ldots \otimes r_n^{\mathcal{I}}(x_{n-1}, y) \right) \Rightarrow r^{\mathcal{I}}(x, y) \geqslant d$
trans(r)	$r^{\mathcal{I}}(x,y) \otimes r^{\mathcal{I}}(y,z) \leqslant r^{\mathcal{I}}(x,z)$
dis(r_1, r_2)	$r_1^{\mathcal{I}}(x,y) = 0$ or $r_2^{\mathcal{I}}(x,y) = 0$
ref(r)	$r^{\mathcal{I}}(x,x) = 1$
irr(r)	$r^{\mathcal{I}}(x,x) = 0$
sym(r)	$r^{\mathcal{I}}(x,y) = r^{\mathcal{I}}(y,x)$
asy(r)	$r^{\mathcal{I}}(x,y) = 0$ or $r^{\mathcal{I}}(y,x) = 0$

for roles). We call such an interpretation a *classical interpretation*.

An ontology is *satisfied* by a fuzzy interpretation $\mathcal{I}$ if all of its axioms are satisfied, as defined in Table 3.

In this case, $\mathcal{I}$ is called a *model* of the ontology. An ontology is *consistent* iff it has a model. In fuzzy extensions of $\mathcal{SROIQ}$, axioms are often allowed to express also strict inequalities ($<$ and $>$). However, in the finitely valued setting an axiom $\langle \alpha > d \rangle$ with $d < 1$ can be expressed as $\langle \alpha \geqslant d_{\text{next}} \rangle$ (and similarly for $<$ and the direct lower neighbour of d).

In the literature, it is also common to find negated role assertions of the form $\langle \neg r(a,b) \bowtie d \rangle$ [4,8]. However, in our setting $\langle \neg r(a,b) \leqslant d \rangle$ is equivalent to an assertion of the form $\langle r(a,b) \geqslant d' \rangle$, and similarly for $\langle \neg r(a,b) \geqslant d \rangle$.

Example 1 Suppose that we have a cloud computing environment consisting of multiple servers with their own internal memory and CPU. To model such an environment, we use

- the individual names: serverA, serverB, memA, memB, cpuA, cpuB;
- the concept names: CPU, Memory, Overused, Server, ServerWithLimitedResources, and ServerWithAvailableResources; and
- the role names: hasPart and isConnectedTo.

The assertional knowledge of this domain is modeled via the ABox $\mathcal{A}$

$$\begin{aligned}&\{\mathsf{Server}(\mathsf{serverA}), \mathsf{CPU}(\mathsf{cpuA}), \mathsf{Memory}(\mathsf{memA}),\\ &\langle \mathsf{Overused}(\mathsf{cpuA}) \geqslant 0.8 \rangle, \mathsf{Overused}(\mathsf{memA}),\\ &\mathsf{hasPart}(\mathsf{serverA}, \mathsf{cpuA}), \mathsf{hasPart}(\mathsf{serverA}, \mathsf{memA}),\\ &\langle \mathsf{ServerWithAvailableResources}(\mathsf{serverB}) \geqslant 0.6 \rangle,\\ &\langle \mathsf{isConnectedTo}(\mathsf{serverA}, \mathsf{serverB}) \geqslant 0.8 \rangle\},\end{aligned}$$

which, for example, states that cpuA is overused with degree at least 0.8, and that the memory memA is also overused with degree 1. The terminological knowledge of this domain can be modeled via a TBox $\mathcal{T}$ containing axioms like

$$\begin{aligned}\langle \mathsf{Server} \sqcap{} & \exists \mathsf{hasPart}.(\mathsf{Overused} \sqcap \mathsf{CPU}) \sqcap \\ & \exists \mathsf{hasPart}.(\mathsf{Overused} \sqcap \mathsf{Memory}) \\ & \sqsubseteq \mathsf{ServerWithLimitedResources} \geqslant 0.8 \rangle,\end{aligned}$$

stating that a server with an overused memory and CPU is a server with limited resources. This implication must hold with a degree of at least 0.8.

It should be noted that the concepts CPU, Memory, and Server and the role hasPart are essentially *crisp*; i.e., they can only take values in $\{0, 1\}$. This information can be easily modeled as part of a fuzzy ontology and handled by the reduction algorithm in [3]. In contrast to this, the concepts ServerWithLimitedResources and Overused have a vague nature; that is, they are fuzzy, and the degree to which a server has limited resources and the degree to which a CPU or a memory card is overused can take values strictly between 0 and 1. The role isConnectedTo is also fuzzy and it is used to declare the connection between two servers. The higher the connection degree between two servers, the larger is the bandwidth they use in their communication.

2.3 Conjunctive Queries

Based on the semantics, other reasoning services than consistency of ontologies can be defined. In this paper, we are interested in conjunctive query answering. We give the definition of classical conjunctive queries next.

Definition 2 *(Conjunctive Query)* Let $\mathsf{N_V}$ be a countably infinite set of *variables* disjoint from $\mathsf{N_C}$, $\mathsf{N_R}$, and $\mathsf{N_I}$. An *atom* is a *concept atom* of the form $A(x)$, a *role atom* of the form $r(x, y)$, or an *equality atom* of the form $x \approx y$, where $x, y \in \mathsf{N_V} \cup \mathsf{N_I}$, $A \in \mathsf{N_C}$, and $r \in \mathsf{N_R}$.

A *(k-ary) conjunctive query (CQ)* q is a statement of the form

$$(x_1, \dots, x_k) \leftarrow \alpha_1, \dots, \alpha_m,$$

where $\alpha_1, \dots, \alpha_m$ are atoms, and $x_1, \dots, x_k$ are (not necessarily distinct) variables occurring in these atoms. We call $x_1, \dots, x_k$ the *distinguished variables* of q. $\mathsf{VarInds}(q)$ denotes the set of all variables and individual names occurring in q. If $k = 0$, we call q a *Boolean conjunctive query*.

Let $\mathcal{I}$ be a classical interpretation, q a Boolean CQ, and $\pi : \mathsf{VarInds}(q) \to \Delta^{\mathcal{I}}$ a function such that $\pi(a) = a^{\mathcal{I}}$ for all $a \in \mathsf{N_I}$. If $\pi(x) \in A^{\mathcal{I}}$, then we write $\mathcal{I} \models^{\pi} A(x)$, and $\mathcal{I} \models^{\pi} r(x, y)$ whenever $(\pi(x), \pi(y)) \in r^{\mathcal{I}}$, and $\mathcal{I} \models^{\pi} x \approx y$ if $\pi(x) = \pi(y)$. If $\mathcal{I} \models^{\pi} \alpha$ for all atoms α in q, we write $\mathcal{I} \models^{\pi} q$ and call π a *match* for $\mathcal{I}$ and q. We say that $\mathcal{I}$ *satisfies* q and write $\mathcal{I} \models q$ if there is a match π for $\mathcal{I}$ and q.

A *(k-ary) union of conjunctive queries (UCQ)* q_{UCQ} is a set of k-ary conjunctive queries. An interpretation $\mathcal{I}$ satisfies a Boolean UCQ q_{UCQ}, written $\mathcal{I} \models q_{\mathrm{UCQ}}$ if $\mathcal{I} \models q$ for some $q \in q_{\mathrm{UCQ}}$. For a Boolean (U)CQ q and an ontology $\mathcal{O}$, we write $\mathcal{O} \models q$ and say that $\mathcal{O}$ *entails* q if $\mathcal{I} \models q$ holds for all models $\mathcal{I}$ of $\mathcal{O}$.

Consider now an arbitrary k-ary (U)CQ q and a k-tuple $\mathfrak{a} \in {\mathsf{N_I}}^k$ of individual names. We say that $\mathfrak{a}$ is an *answer* to q w.r.t. an ontology $\mathcal{O}$ if $\mathcal{O}$ entails the Boolean (U)CQ $\mathfrak{a}(q)$ resulting from q by replacing all distinguished variables according to $\mathfrak{a}$ (and possibly introducing new equality atoms if some of the distinguished variables in an answer tuple are equal).

The problem of *query answering* is to compute all answers of a (U)CQ w.r.t. a given ontology. Query answering can be reduced to query entailment by testing all possible tuples $a \in {\mathsf{N_I}}^k$, which yields an exponential blow-up. It is well known that query entailment and query answering can be mutually reduced and that decidability and complexity results carry over modulo the mentioned blow-up [15].

Example 3 Consider the UCQ $\mathsf{isMonitoredBy}$, consisting of the following CQs:

$$(y, x) \leftarrow \mathsf{monitors}(x, y),$$
$$(x, x) \leftarrow \mathsf{SelfMonitored}(x).$$

To obtain all answers of this UCQ, we consider all possible tuples $(a, b) \in \mathsf{N_I}$ and instantiate the CQs as follows:

$$() \leftarrow \mathsf{monitors}(b, a),$$
$$() \leftarrow \mathsf{SelfMonitored}(a),\ a \approx b,$$

which results in a Boolean UCQ. The latter is entailed by an ontology $\mathcal{O}$ if one can derive that in all models of $\mathcal{O}$ either the assertion $\mathsf{monitors}(b, a)$ holds, or else both $a \approx b$ and $\mathsf{SelfMonitored}(a)$ are satisfied.

If the distinguished variables are clear from the context, we may also omit them and write a CQ simply as a set of atoms.

In fuzzy DLs, conjunctive queries can be of two different types: *threshold conjunctive queries* or *general fuzzy queries* [32,40,41].[4]

Threshold queries ask for tuples of individuals that satisfy a set of assertions to at least some given degree. For example, the threshold query

$$\{\mathsf{Server}(x) \geqslant 1, \mathsf{hasPart}(x, y) \geqslant 1, \mathsf{CPU}(y) \geqslant 1, \mathsf{Overused}(y) \geqslant 0.6\}$$

asks for all pairs of servers and CPUs such that the CPU is a part of the server and is also overused to a degree of at least 0.6.

Definition 4 (*Threshold Conjunctive Query*) A *degree atom* is an expression of the form $\alpha \geqslant d$, where α is an atom and $d \in \mathscr{C}_{>0}$. A *(k-ary) threshold conjunctive query* q_{t} is of the form

$$(x_1, \dots, x_k) \leftarrow \alpha_1 \geqslant d_1, \dots, \alpha_m \geqslant d_m,$$

where $\alpha_1 \geqslant d_1, \dots, \alpha_m \geqslant d_m$ are degree atoms and $x_1, \dots, x_k$ are variables. As before, $\mathsf{VarInds}(q_{\mathrm{t}})$ denotes the set of variables and individuals occurring in the threshold CQ q_{t}.

Let $\mathcal{I}$ be an interpretation, q_{t} a Boolean threshold CQ, and $\pi : \mathsf{VarInds}(q_{\mathrm{t}}) \to \Delta^{\mathcal{I}}$ a function that maps each $a \in \mathsf{N_I}$ to $a^{\mathcal{I}}$. The *degree* of an atom $\alpha = A(x)$ w.r.t. π is defined as $\alpha^{\mathcal{I}}(\pi) := A^{\mathcal{I}}(\pi(x))$, and we set $\alpha^{\mathcal{I}}(\pi) := r^{\mathcal{I}}(\pi(x), \pi(y))$ for $\alpha = r(x, y)$; finally, for $\alpha = x \approx y$ we define $\alpha^{\mathcal{I}}(\pi) := 1$ if $\pi(x) = \pi(y)$, and $\alpha^{\mathcal{I}}(\pi) := 0$ otherwise. If $\alpha^{\mathcal{I}}(\pi) \geqslant d$ holds for all degree atoms $\alpha \geqslant d$ in q_{t}, then we write $\mathcal{I} \models^{\pi} q_{\mathrm{t}}$

[4] In [41], queries are defined that allow for grouping, aggregation, and ranking. Although we do not consider such queries here, we generalize our basic queries in Sect. 4.3.

and call π a *match* for $\mathcal{I}$ and q_t. The notions of satisfaction, entailment, and answers are defined as for classical CQs.

General fuzzy CQs, in contrast, have the same syntax as classical conjunctive queries. Their answers are the tuples of individuals satisfying them to a degree greater than 0, together with the degree to which the query is satisfied. For example,

$$\{\mathsf{Server}(x), \mathsf{hasPart}(x, y), \mathsf{CPU}(y), \mathsf{Overused}(y)\} \tag{3}$$

asks for all overused CPUs that belong to a server, along with the degree to which these CPUs are overused. To obtain the degree of the query from the individual degrees of the atoms, the fuzzy operator interpreting the conjunction is used.

Definition 5 *(Fuzzy Conjunctive Query)* A *(k-ary) fuzzy conjunctive query* q_f is of the form

$$(x_1, \ldots, x_k) \leftarrow \alpha_1, \ldots, \alpha_m,$$

where $\alpha_1, \ldots, \alpha_m$ are atoms and $x_1, \ldots, x_k$ are variables. Let $\mathcal{I}$ be an interpretation, q_f a Boolean fuzzy CQ, and π a mapping as in Definition 4. If $\bigotimes_{\alpha \in q_\mathsf{f}} \alpha^\mathcal{I}(\pi) \geqslant d > 0$, then we write $\mathcal{I} \models^\pi q_\mathsf{f} \geqslant d$ and call π a *match* for $\mathcal{I}$ and q_f with a *degree* of at least d. We say that $\mathcal{I}$ *satisfies* q_f with a degree of at least d and write $\mathcal{I} \models q_\mathsf{f} \geqslant d$ if there is such a match. If $\mathcal{I} \models q_\mathsf{f} \geqslant d$ for all models $\mathcal{I}$ of an ontology $\mathcal{O}$, we write $\mathcal{O} \models q_\mathsf{f} \geqslant d$ and say that $\mathcal{O}$ *entails* q_f with a degree of at least d. Finally, a tuple $\mathfrak{a} \in {\mathsf{N_I}}^k$ is an *answer* to a k-ary fuzzy CQ q_f w.r.t. $\mathcal{O}$ with a degree of at least d if $\mathcal{O}$ entails $\mathfrak{a}(q_\mathsf{f})$ with a degree of at least d.

The query entailment problem for a (Boolean) threshold CQ is to decide whether $\mathcal{O} \models q_\mathsf{t}$. For fuzzy CQs, we may consider two variants of the query entailment problem, namely

- to decide whether $\mathcal{O} \models q_\mathsf{f} \geqslant d$ for a given $d \in \mathscr{C}_{>0}$, or
- to find the best entailment degree $\max\{d \mid \mathcal{O} \models q_\mathsf{f} \geqslant d\}$.

Since we consider only finitely valued semantics over the chain $\mathscr{C}$, these two problems can be polynomially reduced to each other. As for classical query answering, it suffices to analyze the complexity of query *entailment*; the results can then be transferred to query answering [15].

Example 6 Consider the following queries:

$$q_\mathsf{t} := \{\mathsf{hasPart}(x, y) \geqslant 1, \mathsf{Overused}(y) \geqslant 0.9\},$$
$$q_\mathsf{f} := \{\mathsf{hasPart}(x, y), \mathsf{Overused}(y)\},$$

and the ontology from Example 1. An answer to the query q_t is $(\mathsf{serverA}, \mathsf{memA})$, but not $(\mathsf{serverA}, \mathsf{cpuA})$ since cpuA is only overused to degree 0.8. The answers to q_f are the pairs $(\mathsf{serverA}, \mathsf{cpuA})$ with degree $\geqslant 0.8$ and $(\mathsf{serverA}, \mathsf{memA})$ to degree 1.

Remark 7 A threshold CQ with inequalities using $\leqslant$ would correspond to a classical CQ containing negated role atoms, for which query answering is undecidable even in very inexpressive DLs [24,33]. Similarly, upper bounds for fuzzy conjunctive queries q_f, i.e., asking whether $\mathcal{O} \models q_\mathsf{f} \leqslant d$, can be seen as a generalized form of disjunction of (negated) query atoms. For these reasons, we consider only inequalities using $\geqslant$.

Before we turn to answering such queries over fuzzy ontologies, we describe the reduction of expressive finitely valued fuzzy ontologies to classical ones.

3 Reduction of Finitely Valued Fuzzy Ontologies to Classical Ontologies

A popular reasoning technique for fuzzy DLs based on finite chains is the reduction of the fuzzy ontology to a classical one. This allows to use existing DL systems to reason in the fuzzy description logic. However, a major drawback of existing approaches for finite chains using arbitrary t-norms (see [8, 10,28]) is that this reduction introduces an exponential blow-up in the size of the fuzzy ontology. While this handicap can be remedied by our normalization step described in Sect. 3.2 (see also the experiments in Sect. 5.2), another obstacle needs to be addressed first: the reduction proposed in [8,10,28] is not correct for number restrictions.

In the following, we describe this problem in detail and propose a (partial) solution.

3.1 Treating Number Restrictions

The reduction in [8,10,28] is based on the idea to simulate number restrictions by existential restrictions in the following way. For a number restriction $\geq m\, r.C$, the new concept names $B_1, \ldots, B_m$ and the axioms

- $\top \sqsubseteq B_1 \sqcup \cdots \sqcup B_m$ and
- $B_i \sqcap B_j \sqsubseteq \bot$ for all $i, j,\ 1 \leqslant i < j \leqslant m$.

are introduced, which require them to form a partition. Subsequently, the number restriction $\geq m\, r.C$ is replaced by the concept $\exists r.(C \sqcap B_1) \sqcap \ldots \sqcap \exists r.(C \sqcap B_m)$. The following classical example shows that this replacement does not preserve the semantics of the number restrictions, and thus cannot be correct in the fuzzy case, either.

Example 8 Consider the following ABox and TBox:

$$\begin{aligned}\mathcal{A} &:= \{r(a,a),\ r(a,b),\ r(b,a),\ r(b,c),\ r(c,b),\ r(c,c),\\ &\quad a \neq b,\ b \neq c,\ a \neq c\},\\ \mathcal{T} &:= \{\top \sqsubseteq {\leq}2\,r.\top,\ \top \sqsubseteq {\geq}2\,r.\top\}.\end{aligned}$$

A simple model for $\mathcal{O} = (\mathcal{A}, \mathcal{T}, \emptyset)$ is given by $\Delta^{\mathcal{I}} := \{a, b, c\}$ and $r^{\mathcal{I}} := \{(a,a), (a,b), (b,a), (b,c), (c,b), (c,c)\}$. Thus, the ontology $\mathcal{O}$ is consistent. By replacing ${\geq}2\,r.\top$ according to the method described above, we obtain the TBox

$$\begin{aligned}\mathcal{T}' := \{&\top \sqsubseteq {\leq}2\,r.\top,\ \top \sqsubseteq \exists r.B_1 \sqcap \exists r.B_2,\\ &B_1 \sqcap B_2 \sqsubseteq \bot,\ \top \sqsubseteq B_1 \sqcup B_2\}.\end{aligned}$$

We show that the resulting ontology $(\mathcal{A}, \mathcal{T}', \emptyset)$ is inconsistent. Assume to the contrary that there exists a model $\mathcal{I}'$ of the ontology $\mathcal{O}' = (\mathcal{A}, \mathcal{T}', \emptyset)$. Without loss of generality, suppose that it interprets the individual names a, b, c as themselves. Thus, we must have $r^{\mathcal{I}} \subseteq r^{\mathcal{I}'}$. There are only eight possible combinations for a, b, c belonging to either B_1 or B_2. Suppose first that $a, b \in B_1^{\mathcal{I}'}$ and $c \in B_2^{\mathcal{I}'}$. Then, by the axiom $\top \sqsubseteq \exists r.B_1 \sqcap \exists r.B_2$ the individual a must have yet another r-successor $x \in B_2^{\mathcal{I}'}$. However, this contradicts the GCI $\top \sqsubseteq {\leq}2\,r.\top$. Similar arguments apply for all other combinations and, therefore, the ontology is inconsistent.

It should be noted that for Gödel and Zadeh semantics alternative (correct) reductions of number restrictions exist [7,8].

We now propose an alternative encoding of number restrictions when using other fuzzy semantics, avoiding the problem exhibited by Example 8. Intuitively, instead of using a partition of the target concept C of a restriction ${\geq}m\,r.C$, we will partition the role r. Note first that at-most restrictions can be expressed using negation and at-least restrictions; that is, ${\leq}m\,r.C$ has the same semantics as $\neg({\geq}(m+1)\,r.C)$ (cf. Table 2). Hence, in the following, we focus on methods for handling at-least number restrictions. Furthermore, we can assume without loss of generality that they only occur in axioms of the forms

$$\langle A \sqsubseteq {\geq}m\,r.B \geqslant d\rangle \quad \text{and} \quad \langle {\geq}m\,r.B \sqsubseteq A \geqslant d\rangle,$$

where A and B are concept names (cf. Sect. 3.2).

Axioms of the first kind can be equivalently expressed using m fresh role names $r_1, \ldots, r_m$ in the following axioms:

$$\langle A \sqsubseteq \exists r_i.C \geqslant d\rangle, \quad r_i \sqsubseteq r, \quad \mathrm{dis}(r_j, r_k),$$

for all $i, j, k \in \{1, \ldots, m\}$ with $j < k$. This is correct due to the minimum used in the semantics of at-least restrictions. More precisely, every model of the original axiom can be extended by a suitable interpretation of the new role names to a model of the resulting axioms, and every model of the latter is immediately a model of the former. Hence, we can eliminate all at-least restrictions that occur on the right-hand side of GCIs (and all at-most restrictions that occur on the left-hand side of GCIs).

Unfortunately, this approach does not work for at-least restrictions occurring on the left-hand side of GCIs. The reason is that the presence of m many r-successors satisfying C does not imply that these successors can be reached using one of the disjoint roles $r_1, \ldots, r_m$. However, this can be expressed using the additional role axiom

$$r \sqsubseteq r_1 \sqcup \cdots \sqcup r_m, \tag{4}$$

which involves a *role disjunction* that is interpreted using the maximum, i.e.,

$$(r_1 \sqcup \cdots \sqcup r_m)^{\mathcal{I}}(x, y) := \max_{i=1}^{m} r_i^{\mathcal{I}}(x, y).$$

Role disjunction is an example of a (safe) Boolean role constructor, which can be added to most classical DLs without increasing the complexity of reasoning [34]. Moreover, some query answering procedures for classical DLs even work in the presence of such constructors [16]. Unfortunately, to the best of our knowledge, role disjunctions are not yet supported by any classical DL reasoner.

In the presence of axiom (4) and the role disjointness axioms from above, the GCI $\langle {\geq}m\,r.C \sqsubseteq A \geqslant d\rangle$ can now be equivalently expressed as:

$$\langle \exists r_1.C \sqcap \cdots \sqcap \exists r_m.C \sqsubseteq A \geqslant d\rangle.$$

Unlike the incorrect reduction for number restrictions that was first proposed in [10], our approach does not partition the *range* of the role r in the number restriction, but rather the role itself, and hence it correctly treats the case where a domain element is an r-successor of two different elements that are subject to the same number restriction on r (recall Example 8). However, like the approach of [8,10,28], this incurs an exponential blow-up in the largest number occurring in number restrictions, if these numbers are represented in the ontology using a binary encoding. The reduction is polynomial if we assume unary encoding of numbers.

Since role disjunctions are not supported by $\mathcal{SROIQ}$ or OWL 2, we will restrict the following investigation to *unqualified* number restrictions of the form ${\geq}m\,r := {\geq}m\,r.\top$ and ${\leq}m\,r := {\leq}m\,r.\top$, i.e., to the fuzzy logic $\mathscr{C}$-$\mathcal{SROIN}$. However, we want to emphasize that we can easily treat qualified number restrictions in the following reduction if the classical target language supports role disjunctions. It is straightforward to extend the reduction to deal even with $\mathcal{SROIQB}_s$, the extension of $\mathcal{SROIQ}$ with full Boolean role expressions (which satisfy a safety condition) [34]; the reduction of the role constructors is similar to the one for concepts.

Table 4 Normalization rules for $\mathscr{C}$-$\mathcal{SROIN}$ ontologies

$\langle C \sqsubseteq D \geqslant d\rangle \rightsquigarrow \langle C \sqsubseteq A_D \geqslant d\rangle,\ A_D \sqsubseteq D$	$\langle C(a) \geqslant d\rangle \rightsquigarrow \langle A_C(a) \geqslant d\rangle,\ A_C \sqsubseteq C$
$\langle A \sqsubseteq C \sqcap D \geqslant d\rangle \rightsquigarrow \langle A \sqsubseteq A_C \sqcap A_D \geqslant d\rangle,\ A_C \sqsubseteq C,\ A_D \sqsubseteq D$	$\langle C \sqcap D \sqsubseteq A \geqslant d\rangle \rightsquigarrow \langle A_C \sqcap A_D \sqsubseteq A \geqslant d\rangle,\ C \sqsubseteq A_C,\ D \sqsubseteq A_D$
$\langle A \sqsubseteq C \sqcup D \geqslant d\rangle \rightsquigarrow \langle A \sqsubseteq A_C \sqcup A_D \geqslant d\rangle,\ A_C \sqsubseteq C,\ A_D \sqsubseteq D$	$\langle C \sqcup D \sqsubseteq A \geqslant d\rangle \rightsquigarrow \langle A_C \sqcup A_D \sqsubseteq A \geqslant d\rangle,\ C \sqsubseteq A_C,\ D \sqsubseteq A_D$
$\langle A \sqsubseteq \neg C \geqslant d\rangle \rightsquigarrow \langle A \sqsubseteq \neg A_C \geqslant d\rangle, C \sqsubseteq A_C$	$\langle \neg C \sqsubseteq A \geqslant d\rangle \rightsquigarrow \langle \neg A_C \sqsubseteq A \geqslant d\rangle,\ A_C \sqsubseteq C$
$\langle A \sqsubseteq \exists r.C \geqslant d\rangle \rightsquigarrow \langle A \sqsubseteq \exists r.A_C \geqslant d\rangle,\ A_C \sqsubseteq C$	$\langle \exists r.C \sqsubseteq A \geqslant d\rangle \rightsquigarrow \langle \exists r.A_C \sqsubseteq A \geqslant d\rangle,\ C \sqsubseteq A_C$
$\langle A \sqsubseteq \forall r.C \geqslant d\rangle \rightsquigarrow \langle A \sqsubseteq \forall r.A_C \geqslant d\rangle,\ A_C \sqsubseteq C$	$\langle \forall r.C \sqsubseteq A \geqslant d\rangle \rightsquigarrow \langle \forall r.A_C \sqsubseteq A \geqslant d\rangle,\ C \sqsubseteq A_C$
$\langle r_1r_2r_3 \ldots r_m \sqsubseteq r \geqslant d\rangle \rightsquigarrow \langle r_{r_1r_2}r_3 \ldots r_m \sqsubseteq r \geqslant d\rangle,\ r_1r_2 \sqsubseteq r_{r_1r_2}$	

3.2 Ontology Normalization for $\mathscr{C}$-$\mathcal{SROIN}$ Ontologies

The reason that the reductions described in [7,8,10] can cause an exponential blow-up in the size of the ontology is that concept constructors may be nested to arbitrary depths. In this subsection, we propose a normalization step to ensure that each GCI and concept assertion contains at most one concept constructor, and that each complex role inclusion contains at most two roles on the left-hand side. Because of this, the subsequent reduction of a $\mathscr{C}$-$\mathcal{SROIN}$ ontology $\mathcal{O}$ to a classical $\mathcal{SROIN}$ ontology $\mathcal{O}_\mathsf{c}$ causes only a linear blow-up in the size of $\mathcal{O}$ (and a quadratic blow-up in the size of $\mathscr{C}$). For an experimental evaluation of the resulting difference in ontology size and reasoning performance, see Sect. 5.2.

The normalization proceeds by exhaustively replacing each axiom by a set of axioms according to Table 4. In that table, A, A_C, A_D denote concept names, $\top$, or $\bot$; C, D are complex concepts that are neither concept names, $\top$, nor $\bot$; and $r_1, \ldots, r_m$, r, and $r_{r_1r_2}$ are roles. A_C and A_D are fresh concept names that abbreviate the concepts C and D, respectively. In the last rule, $r_{r_1r_2}$ is a fresh role name that stands for the role composition of r_1 and r_2. For simplicity, we have given the rules for conjunctions and disjunctions only for the case where both operands are complex concepts. However, if only one of them is a complex concept, we would not introduce a new concept name for the other operand. Note that nominals, unqualified number restrictions, and local reflexivity concepts do not need to be normalized.

It should be noted that this reduction is not correct under Zadeh semantics due to the properties of the implication function. However, [4] provides a different reduction for this case that does not exhibit an exponential blow-up even without normalization. Hence, we consider in the following result only semantics that are based on finitely valued t-norms and their induced operators $\Rightarrow$, $\ominus$, and $\oplus$.

Proposition 9 *Let $\mathcal{O}'$ be the ontology resulting from the exhaustive application of the rules in Table 4 to a $\mathscr{C}$-$\mathcal{SROIN}$ ontology $\mathcal{O}$.*

Under t-norm-based semantics, every model of $\mathcal{O}$ can be extended to a model of $\mathcal{O}'$ by interpreting the new concept names A_C like C and $r_{r_1r_2}$ like r_1r_2.[5] *Moreover, every model of $\mathcal{O}'$ is already a model of $\mathcal{O}$.*

This simple observation immediately shows that $\mathcal{O}'$ is consistent iff $\mathcal{O}$ is consistent. Moreover, it allows us to prove correctness of the normalization procedure also with respect to the other reasoning tasks we will consider in the following sections. Furthermore, it is easy to see that the normalization could be extended to deal also with qualified number restrictions ($\mathcal{Q}$).

While this procedure involves the introduction of linearly many new concept names, it allows us to circumvent the exponential blow-up exhibited by previous reductions.

Remark 10 The reason why this normalization reduces the complexity of the following reduction is that it ensures that each axiom contains at most three occurrences of concept or role names. However, we will see in the following subsection that concept and role names that are interpreted classically, i.e., can take only the values 0 and 1, do not take part in the reduction (see also [2,5]). Hence, it is enough to ensure that each axiom contains at most three occurrences of *fuzzy* concept or role names. Such axioms do not need to be reduced any further. Nevertheless, all complexity results concerning the reduction in the following section remain valid.

Example 11 The normalized form of the TBox $\mathcal{T}$ containing the GCI from Example 1 is as follows:

$$\begin{aligned}
\{&\mathsf{Overused} \sqcap \mathsf{CPU} \sqsubseteq A,\\
&\mathsf{Overused} \sqcap \mathsf{Memory} \sqsubseteq B,\\
&\exists \mathsf{hasPart}.A \sqsubseteq C,\\
&\exists \mathsf{hasPart}.B \sqsubseteq D,\\
&\mathsf{Server} \sqcap C \sqsubseteq E,\\
&\langle E \sqcap D \sqsubseteq \mathsf{ServerWithLimitedResources} \geqslant 0.8\rangle\}
\end{aligned}$$

However, since the GCI in $\mathcal{T}$ contains only three occurrences of names of fuzzy concepts (two times Overused and once ServerWithLimitedResources), we can use $\mathcal{T}$ as it is in the following reduction.

[5] Where $(r_1r_2)^\mathcal{I}(x,z) := \sup_{y\in\Delta^\mathcal{I}} r_1^\mathcal{I}(x,y) \otimes r_2^\mathcal{I}(y,z)$ (cf. Table 3).

$$\begin{aligned}\mathcal{A}_{\mathsf{c}} := \{&\mathsf{Server}(\mathsf{serverA}),\ \mathsf{CPU}(\mathsf{cpuA}),\ \mathsf{Memory}(\mathsf{memA}),\ \mathsf{Overused}_{\geqslant 0.8}(\mathsf{cpuA}),\ \mathsf{Overused}_{\geqslant 1}(\mathsf{memA}),\ \mathsf{hasPart}(\mathsf{serverA},\mathsf{cpuA}),\\ &\mathsf{hasPart}(\mathsf{serverA},\ \mathsf{memA}),\ \mathsf{ServerWithAvailableResources}_{\geqslant 0.6}(\mathsf{serverB}),\ \mathsf{isConnectedTo}_{\geqslant 0.8}(\mathsf{serverA},\ \mathsf{serverB})\}\end{aligned}$$

$$\mathcal{T}_{\mathsf{c}} := \{\mathsf{Overused}_{\geqslant 0.4} \sqsubseteq \mathsf{Overused}_{\geqslant 0.2}, \dots\} \cup \bigcup_{\substack{d_1, d_2 \text{ minimal, } d_3 \text{ maximal}\\ \text{such that } (d_1 \otimes d_2) \Rightarrow d_3 < 0.8}} \{\mathsf{Server} \sqcap \exists \mathsf{hasPart}.(\mathsf{Overused}_{\geqslant \mathsf{d}_1} \sqcap \mathsf{CPU}) \sqcap \exists \mathsf{hasPart}.(\mathsf{Overused}_{\geqslant \mathsf{d}_2} \sqcap \mathsf{Memory}) \sqsubseteq \mathsf{ServerWithLimitedResources}_{>\mathsf{d}_3}\}$$

Fig. 1 ABox and TBox for Example 14

We will assume in the following that $\mathcal{O}$ is already normalized. The remainder of the reduction is very similar to the one described in [8] (except for number restrictions).

3.3 The Reduction Algorithm

Each concept name and role name in $\mathcal{O}$ is mapped onto a set of concepts and roles corresponding to their *α-cuts*, which are crisp sets containing all elements that belong to a fuzzy set to at least a given degree α. For example, if the concept name $\mathsf{Overused}$ describes the degree to which a CPU is overused, then $\mathsf{Overused}_{\geqslant 0.6}$ represents the set of CPUs that are overused to a degree of at least 0.6. It is clear that we do not need to consider the value 0 for such cuts, as $A_{\geqslant 0}$ always describes the whole domain. We may also refer to concept names of the form $A_{>d}$ for $d \in \mathscr{C}$ and $d < 1$, which is a short-hand notation for $A_{\geq d_{\text{next}}}$, and similarly for role names.

The $\mathcal{SROIN}$ ontology $\mathcal{O}_{\mathsf{c}}$ obtained from the reduction has the following form:

- To preserve the semantics of α-cuts of concept and role names, the following axioms are added to $\mathcal{O}_{\mathsf{c}}$ for all $A \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, and $d \in \mathscr{C}$ with $0 < d < 1$:

  $$A_{>d} \sqsubseteq A_{\geqslant d},\ \ r_{>d} \sqsubseteq r_{\geqslant d}.$$

- Each complex concept C appearing in $\mathcal{O}$ is mapped to the complex concept $\rho(C, \geqslant d)$ that represents its α-cut regarding degree d, as defined in the first part of Table 7 in the appendix.
- Each axiom in $\mathcal{O}$ is then mapped to a classical axiom or set of axioms in $\mathcal{O}_{\mathsf{c}}$ according to the mapping κ defined in the second part of Table 7.

For a more detailed analysis of the reduction rules, the interested reader may refer to [7,8,10].

We provide a detailed proof of correctness in the appendix.

Theorem 12 *Let $\mathcal{O}$ be a $\mathscr{C}$-$\mathcal{SROIN}$ ontology. Then, $\mathcal{O}$ has a fuzzy model iff its reduced form $\mathcal{O}_{\mathsf{c}}$ has a classical model.*

Our normalization procedure allows us to show the following improved complexity bounds. The proof of the following lemma can be found in Appendix 1.

Lemma 13 *For a normalized $\mathscr{C}$-$\mathcal{SROIN}$ ontology $\mathcal{O}$, the size of $\mathcal{O}_{\mathsf{c}}$ is linear in the size of $\mathcal{O}$ and quadratic in the size of $\mathscr{C}$.*

This means that, by simply introducing the normalization step, we can avoid the exponential blow-up of the crispification approach. In particular, we greatly improve the exponential bounds shown in [8,10].

Example 14 Figure 1 contains the reduced form of the ontology from Example 1 w.r.t. Łukasiewicz semantics over the chain with six elements $\mathscr{C} = \{0, 0.2, 0.4, 0.6, 0.8, 1\}$. We have taken into account that one does not need to consider α-cuts of classical concept and role names. This nicely illustrates how classical concepts and roles help to reduce the size of the reduction [2,5]. Not only do we have 1 crisp concept instead of $n - 1$ cut concepts, but the number of disjunctions and conjunctions introduced can be reduced dramatically (cf. Table 7).

4 Conjunctive Query Answering for Fuzzy DLs

In this section, we show how to solve the problem of answering threshold and fuzzy CQs in finitely valued fuzzy DLs by taking advantage of existing algorithms for answering unions of conjunctive queries in classical DLs. Our solution is based on the reduction technique described in Sect. 3 .

4.1 Translating Fuzzy and Threshold CQs

In the following, we define a function κ that maps each threshold CQ and fuzzy CQ to a (U)CQ in a classical DL. The idea is that we can then evaluate these classical queries over $\mathcal{O}_{\mathsf{c}}$ to answer the original queries over $\mathcal{O}$. The shape of the mapping κ depends on the type of query, and uses α-cuts for reducing fuzzy concept and role names to classical ones (cf. Sect. 3).

We first define the function κ for degree atoms of queries analogously as it was done for assertions:

$$\kappa(A(x) \geqslant d) := A_{\geqslant d}(x),$$
$$\kappa(r(x, y) \geqslant d) := r_{\geqslant d}(x, y).$$

This definition is then lifted to threshold CQs q_t in the obvious way: $\kappa(q_\mathsf{t}) := \{\kappa(\alpha) \mid \alpha \in q_\mathsf{t}\}$.

For transforming fuzzy CQs, we use α-cuts as in the translation of concepts. Thus, in the case of fuzzy CQs, κ receives as input also a membership degree from $\mathscr{C}_{>0}$, which will be a lower bound for the degree of the query. Recall that the answers depend on the operator $\otimes$ that interprets the conjunction in the logic under consideration. Since there can be several different combinations of degrees for the different atoms that result in the same degree, fuzzy CQs are translated into unions of (classical) conjunctive queries, representing each of these combinations.

Formally, for a fuzzy CQ $q_\mathsf{f} = \{\alpha_1, \ldots, \alpha_n\}$ and $d \in \mathscr{C}_{>0}$, $\kappa(q_\mathsf{f}, \geqslant d)$ is the set of all conjunctive queries

$$\{\kappa(\alpha_1 \geqslant d_1), \ldots, \kappa(\alpha_n \geqslant d_n)\},$$

where $d_1, \ldots, d_n \in \mathscr{C}$ are such that $\bigotimes_{i=1}^n d_i \geqslant d$.

Example 15 Consider a threshold CQ asking for all pairs of connected servers such that the first one has limited and the second one has available resources:

$$\begin{aligned}&\{\mathsf{ServerWithLimitedResources}(x) \geqslant 0.8,\\ &\quad \mathsf{isConnectedTo}(x, y) \geqslant 0.6,\\ &\quad \mathsf{ServerWithAvailableResources}(y) \geqslant 0.6\}.\end{aligned}$$

This threshold CQ is reduced to the following classical CQ:

$$\begin{aligned}&\{\mathsf{ServerWithLimitedResources}_{\geqslant 0.8}(x),\\ &\quad \mathsf{isConnectedTo}_{\geqslant 0.6}(x, y),\\ &\quad \mathsf{ServerWithAvailableResources}_{\geqslant 0.6}(y)\}.\end{aligned}$$

A fuzzy CQ asking for the same information, but without the thresholds, is the following:

$$\begin{aligned}&\{\mathsf{ServerWithLimitedResources}(x),\\ &\quad \mathsf{isConnectedTo}(x, y),\\ &\quad \mathsf{ServerWithAvailableResources}(y)\}.\end{aligned}$$

To acquire all the pairs (x, y) that satisfy this query with degree at least 0.8, for the Łukasiewicz t-norm over the chain with 6 membership degrees, the query is reduced to the following union of classical CQs:

$$\begin{aligned}\big\{&\{\mathsf{ServerWithLimitedResources}_{\geqslant 0.8}(x),\\ &\quad \mathsf{isConnectedTo}_{\geqslant 1}(x, y),\\ &\quad \mathsf{ServerWithAvailableResources}_{\geqslant 1}(y)\},\\ &\{\mathsf{ServerWithLimitedResources}_{\geqslant 1}(x),\\ &\quad \mathsf{isConnectedTo}_{\geqslant 0.8}(x, y),\\ &\quad \mathsf{ServerWithAvailableResources}_{\geqslant 1}(y)\},\\ &\{\mathsf{ServerWithLimitedResources}_{\geqslant 1}(x),\\ &\quad \mathsf{isConnectedTo}_{\geqslant 1}(x, y),\\ &\quad \mathsf{ServerWithAvailableResources}_{\geqslant 0.8}(y)\}\big\}.\end{aligned}$$

The following theorem states that our query reduction is sound and complete.

Theorem 16 *Let $\mathcal{O}_\mathsf{c}$ be the classical version of the fuzzy ontology $\mathcal{O}$, q_t be a threshold* CQ*, q_f be a fuzzy* CQ*, and $d \in \mathscr{C}$. Then, the following equivalences hold:*

1. $\mathcal{O} \models q_\mathsf{t} \;\Leftrightarrow\; \mathcal{O}_\mathsf{c} \models \kappa(q_\mathsf{t})$
2. $\mathcal{O} \models q_\mathsf{f} \geqslant d \;\Leftrightarrow\; \mathcal{O}_\mathsf{c} \models \kappa(q_\mathsf{f}, \geqslant d)$.

Proof To prove that $\mathcal{O}_\mathsf{c} \models \kappa(q_\mathsf{t})$ implies $\mathcal{O} \models q_\mathsf{t}$, consider any fuzzy model $\mathcal{I}$ of $\mathcal{O}$. By Proposition 9, $\mathcal{I}$ can be extended to a model of the normalized ontology $\mathcal{O}'$. We now define the classical interpretation $\mathcal{J} = \{\Delta^\mathcal{J}, \cdot^\mathcal{J}\}$ as follows (cf. Appendix 1):

$$\begin{aligned}\Delta^\mathcal{J} &:= \Delta^\mathcal{I}\\ a^\mathcal{J} &:= a^\mathcal{I}\\ A^\mathcal{J}_{\geqslant d} &:= \left\{\beta \mid A^\mathcal{I}(\beta) \geqslant d\right\}\\ r^\mathcal{J}_{\geqslant d} &:= \left\{(\beta, \gamma) \mid r^\mathcal{I}(\beta, \gamma) \geqslant d\right\}.\end{aligned} \tag{5}$$

By Lemma 22, $\mathcal{J}$ is a classical model of $\mathcal{O}_\mathsf{c}$. Since $\mathcal{J} \models \mathcal{O}_\mathsf{c}$ and $\mathcal{O}_\mathsf{c} \models \kappa(q_\mathsf{t})$, it follows that $\mathcal{J} \models \kappa(q_\mathsf{t})$. By the construction of $\mathcal{J}$, it can be easily verified that $\mathcal{I} \models q_\mathsf{t}$.

It can be shown in a similar way that $\mathcal{O} \models q_\mathsf{f} \geqslant d$ implies $\mathcal{O}_\mathsf{c} \models \kappa(q_\mathsf{f}, \geqslant d)$. To prove the opposite direction we build for each classical model $\mathcal{J}$ of $\mathcal{O}_\mathsf{c}$, the fuzzy interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where

$$\begin{aligned}\Delta^\mathcal{I} &:= \Delta^\mathcal{J}\\ a^\mathcal{I} &:= a^\mathcal{J}\\ A^\mathcal{I}(\beta) &:= \max\left\{d \mid \beta \in A^\mathcal{J}_{\geqslant d}\right\}\\ r^\mathcal{I}(\beta, \gamma) &:= \max\left\{d \mid (\beta, \gamma) \in r^\mathcal{J}_{\geqslant d}\right\}.\end{aligned} \tag{6}$$

(cf. Appendix 1). By Proposition 9 and Lemma 20, $\mathcal{I}$ is a fuzzy model of the original ontology $\mathcal{O}$.

It is straightforward to show that $\mathcal{J} \models \kappa(q_\mathsf{t})$ whenever $\mathcal{I} \models q_\mathsf{t}$, and similarly for fuzzy CQs. □

It is easy to see that this result applies not only to the reduction described in Sect. 3, but also to any reduction that can be shown correct using the definitions in (5) and (6), e.g., the one optimized for $\mathscr{C}$-$\mathcal{SROIQ}$ under Gödel and Zadeh semantics in [8].

4.2 Complexity Results

By Lemma 13, the size of the crispified ontology $\mathcal{O}_\mathsf{c}$ is polynomial in the size of $\mathcal{O}$. Therefore, we can transfer all complexity results for answering classical CQs over classical sublogics of $\mathcal{SROIN}$ directly to the query answering problem for threshold CQs. In particular, recall that CQ entailment in $\mathcal{SROIN}$ can be decided in 3- ExpTime [16], is 2- ExpTime-complete for $\mathcal{SHIN}$, and ExpTime-complete for $\mathcal{ALCHN}$ [20,27].

Moreover, for fuzzy DLs where correct crispification algorithms exist even for $\mathcal{SROIQ}$, e.g., for Gödel and Zadeh semantics [8] or in the presence of role disjunctions (see Sect. 3.1), any classical CQ answering technique that is able to handle number restrictions can be applied also for threshold CQs. Under this condition, answering threshold CQs in finitely valued extensions of $\mathcal{SHIQ}$ and $\mathcal{SHOQ}$ is 2-ExpTime-complete [20,22], for $\mathcal{SROQ}$ and $\mathcal{SRIQ}$ it can be done in 3- ExpTime [16], while for $\mathcal{SHQ}$ it becomes ExpTime-complete if queries are restricted to *simple* roles, i.e., roles that do not have transitive subroles [27]. Notice, however, that none of these query answering approaches has been implemented so far.

For fuzzy CQs, the complexity increases by an exponential factor due to the blow-up in the translation κ. It is possible to eliminate this blow-up, however, when the minimum t-norm is used, i.e., under Gödel and Zadeh semantics. Then, we can obviously define $\kappa(q_\mathsf{f}, \geqslant d) := \{\kappa(\alpha \geqslant d) \mid \alpha \in q_\mathsf{f}\}$ for any fuzzy CQ q_f, and thus obtain the same complexity results as for threshold CQs.

It should also be noted that the *data complexity* of all these problems is the same as that for classical CQs, as the size of the ABox is not increased by the reduction from $\mathcal{O}$ to $\mathcal{O}_\mathsf{c}$. Since for many applications the TBox remains unchanged, while the ABox changes frequently; the reduction to the crisp ontology $\mathcal{O}_\mathsf{c}$ need not be computed when queries are answered, but need to be computed only once “off-line” beforehand.

4.3 Generalizing the Query Component

So far, we have examined the reduction technique for answering threshold CQs and fuzzy CQs. These two types of queries are immediate extensions of classical CQs, and usually considered in the literature. Nevertheless, the existence of degrees may lead to more general forms of fuzzy CQs in which the score of a query is computed via a monotone scoring function, as described in the following.

Definition 17 A (*k-ary*) *scoring query* q_s is an expression of the form

$$(x_1, \dots, x_k) \leftarrow f(\alpha_1, \dots, \alpha_n),$$

where f is a monotonically increasing *scoring function* with n arguments, $\alpha_1, \dots, \alpha_n$ are atoms, and $x_1, \dots, x_k$ are variables. Let $\mathcal{I}$ be an interpretation, q_s a Boolean scoring query, and π a mapping as in Definition 4. If

$$f(\alpha_1^\mathcal{I}(\pi), \dots, \alpha_n^\mathcal{I}(\pi)) \geqslant d,$$

then we write $\mathcal{I} \models^\pi q_\mathsf{s} \geqslant d$ and call π a *match* for $\mathcal{I}$ and q_s with a *score* of at least d. We say that $\mathcal{I}$ *satisfies* q_s with a score of at least d and write $\mathcal{I} \models q_\mathsf{s} \geqslant d$ if there is such a match. If $\mathcal{I} \models q_\mathsf{s} \geqslant d$ for all models $\mathcal{I}$ of an ontology $\mathcal{O}$, we write $\mathcal{O} \models q_\mathsf{s} \geqslant d$ and say that $\mathcal{O}$ *entails* q_s with a score of at least d. Finally, a tuple $\mathfrak{a} \in \mathsf{N_I}^k$ is an *answer* to a k-ary scoring query q_s w.r.t. $\mathcal{O}$ with a score of at least d if $\mathcal{O}$ entails $\mathfrak{a}(q_\mathsf{s})$ with a score of at least d.[6]

It should be noted that the score may take an arbitrary value in $\mathbb{R}$. This kind of queries has already been considered in the literature [32,40,41]. Fuzzy CQs can be seen as special scoring queries of the form $\alpha_1 \otimes \dots \otimes \alpha_n$. Since $\mathscr{C}$ is finite, the same technique as for fuzzy CQs can be applied here, i.e., considering all possible combinations of degrees in $\mathscr{C}$.

Example 18 Suppose that we are interested in finding all servers that have overused CPU and memory, but the excessive use of CPU should be considered of greater importance than the use of memory. To achieve this, we formulate the following query to include a weighting factor on the degrees of overuse for the different components. For instance, we can use the query

$$\begin{aligned} &\mathsf{Server}(x) \cdot \mathsf{hasPart}(x, y) \cdot \mathsf{CPU}(y) \cdot \mathsf{hasPart}(x, z) \cdot \\ &\quad \mathsf{Memory}(z) \cdot \frac{3 \cdot \mathsf{Overused}(y) + 2 \cdot \mathsf{Overused}(z)}{5} \end{aligned} \tag{7}$$

where the fraction in the last factor takes into account the degrees of overuse of CPU and memory with weights 0.6 and 0.4, respectively.

Assume that $\mathscr{C} = \{0, 0.25, 0.5, 0.75, 1\}$ and that the concepts Server, CPU, and Memory and the role hasPart behave classically. If we want to find all answers that satisfy this query to degree at least 0.25, then we can translate it into a union of classical conjunctive queries that contains, e.g.,

$$\begin{aligned} \{&\mathsf{Server}_{\geqslant 1}(x),\ \ \mathsf{hasPart}_{\geqslant 1}(x, y), \mathsf{CPU}_{\geqslant 1}(y), \\ &\mathsf{hasPart}_{\geqslant 1}(x, z), \mathsf{Memory}_{\geqslant 1}(z), \\ &\mathsf{Overused}_{\geqslant 0.25}(y), \mathsf{Overused}_{\geqslant 0.75}(z)\}. \end{aligned}$$

When evaluated over the reduced ontology, this query returns all triples of elements from Server, CPU, and Memory,

[6] Possibly new equality atoms $a \approx b$ introduced by the instantiation can be connected with a multiplication to the score of the original query (see Example 3).

respectively, where the CPU is overused to degree 0.25 and the memory is overused to degree 0.75. These tuples satisfy the query (7) with a degree of at least

$$0.6 \cdot 0.25 + 0.4 \cdot 0.75 = 0.45 \geqslant 0.25,$$

as desired.

Another interesting problem, specific to weighted logics, is the top-k query answering problem presented in [39–41]. This variation of the fuzzy query answering problem focuses on the k answers with the highest degrees of satisfaction. In a naive approach to solve this problem, the translation function κ for fuzzy or scoring CQs can be iteratively applied starting from the highest to the lowest degrees in $\mathscr{C}$ until the limit of k answers is reached. It has to be investigated if a more sophisticated approach can be adopted to solve this problem.

5 Practical Evaluation

We have proved that, by introducing the normalization step, we can avoid the exponential blow-up of the earlier crispification approach (Sect. 3). We have additionally extended the reduction process to handle the problem of CQ answering (Sect. 4). The main objectives of this section are to

1. Evaluate how the ontology normalization preprocessing improves the total execution time for CQ answering, and
2. Study the practical limitations of the reduction approach for the problem of CQ answering in the finitely valued fuzzy setting.

Rather than implementing a full fuzzy CQ answering system from scratch, we have modified the existing reasoner DeLorean [5].[7] DeLorean is a reduction-based fuzzy DL reasoner that supports fuzzy variants of the description logics $\mathcal{SROIQ}(\mathcal{D})$ and $\mathcal{SHOIN}(\mathcal{D})$ under the finite Gödel, finite Łukasiewicz, and Zadeh semantics. We use DeLorean to transform an input fuzzy ontology into a crisp ontology through the transformation rules described earlier in this paper. Additionally, we have implemented the reduction for threshold queries described in Sect. 4. After the crisp ontology and query are obtained, an arbitrary crisp reasoner can be employed for query answering. For our experiments, we used the PAGOdA system[8] [42,43]. See [2] for an evaluation of the performance of DeLorean on classical reasoning problems in tractable DLs.

Other reasoners that support query answering for expressive DLs could also have been adopted. However, most other systems can only correctly answer CQs that can be directly expressed in the DL, e.g., tree-shaped CQs or instance queries, or employ the simplifying restriction that the non-distinguished variables can only be bound to named individuals [31]. Although in theory PAGOdA also depends on the query answering capabilities of HermiT [21], it was shown in [42] that it correctly answers the standard queries of the LUBM benchmark (which we use in the following), without having to rely on HermiT. This is also the case for the normalized, fuzzified, and crispified versions of the LUBM TBox we use in the following experiments.

5.1 Test Data and Test Set-up

To evaluate our approach, we used the LUBM[9] ontology benchmark [23]. The LUBM benchmark contains terminological knowledge that describes the domain of a university, and generates synthetic OWL data (ABox) over the ontology specifying individuals belonging to the university. The ABox corresponding to a single university contains approximately 1300 concept assertions and 2450 role assertions and can be scaled by a factor of k, producing information for k different universities. In our experiments, we used different ABoxes covering 1, 15, and 30 universities, respectively.

To use LUBM as a benchmark for fuzzy reasoning, we have extended all ABox axioms with random degrees chosen from a fixed finite chain of cardinality 3, 7, or 11, which are interpreted using Łukasiewicz semantics. The axioms in the TBox are always required to hold with degree 1. Note that the original LUBM TBox is formulated in $\mathcal{ELHI}_{R^+}$ (we ignored all datatype axioms), and hence the same holds for our fuzzified variants. However, due to the introduction of disjunctions in the reduction to a classical ontology, the resulting TBox is formulated in $\mathcal{SHI}$ (see Table 7 and [2]).

The queries of the LUBM benchmark were translated to threshold queries by asking each atom to hold with the smallest non-zero degree ($\frac{1}{2}$, $\frac{1}{6}$, and $\frac{1}{10}$ for 3, 7, and 11 truth degrees, respectively).

For all experiments, we report the average runtime over 5 separate runs; the coefficient of variation was mostly below 25 %, which means that the runtimes did not differ much between runs. All the tests were performed on a machine powered by an Intel Core i7 2.6 GHz processor and equipped with 8 GB DDR3 1600MHz main memory.

5.2 Evaluating the Normalization and Its Effect on Query Answering

For the first experiment, we applied the normalization from Sect. 3.2 to the LUBM TBox, and compared the query answering times for a single challenging threshold CQ (based

[7] http://webdiis.unizar.es/~fbobillo/delorean.

[8] PAGOdA version from 23rd of April 2015 commit id: 30b5afef93 in https://github.com/yujiaoz/PAGOdA.

[9] http://swat.cse.lehigh.edu/projects/lubm/.

Table 5 Time needed for preprocessing by PAGOdA (in s) and query execution time (in ms) for instance queries over the original LUBM TBox vs. its normalized version

♯ Degrees	Normalized	TBox size	♯ Universities					
			1		15		30	
			Preprocessing (in s)	Querying (in ms)	Preprocessing (in s)	Querying (in ms)	Preprocessing (in s)	Querying (in ms)
3	no	493	8.94	2.8	9.24	9.8	9.52	18.8
3	yes	518	8.92	2.4	9.21	9.4	9.52	16.6
7	no	2187	13.09	2.4	13.58	5.0	14.25	7.2
7	yes	2022	12.21	2.2	12.57	3.8	13.06	6.4
11	no	5177	26.62	2.0	27.36	3.2	28.45	5.0
11	yes	3942	18.69	2.2	19.60	3.6	20.50	5.6

on query #9 of the LUBM benchmark). The results of this evaluation, based on the Łukasiewicz semantics, are presented in Table 5.

The table compares the sizes of the crispified TBoxes for both the original TBox and the normalized one (counted as the number of occurrences of concept and role names), as well as the running times of PAGOdA. It shows both the time needed by PAGOdA for its own preprocessing of the ontology and the time needed to perform query answering. The preprocessing clearly dominates the overall runtime. In comparison to this, the time needed by PAGOdA to actually answer the queries is almost negligible.

The preprocessing time is mostly determined by the normalization and by the number of degrees, i.e., the size of $\mathscr{C}$, but it also increases slightly with increased ABox size. One can also see that, although the query answering times are not much affected, the size of the TBox, and hence the time PAGOdA needs for preprocessing, can be reduced significantly by normalization. This can also be seen in the graph in Fig. 2, where the normalized variant of the TBox is clearly smaller w.r.t. increasing numbers of degrees than the original TBox.

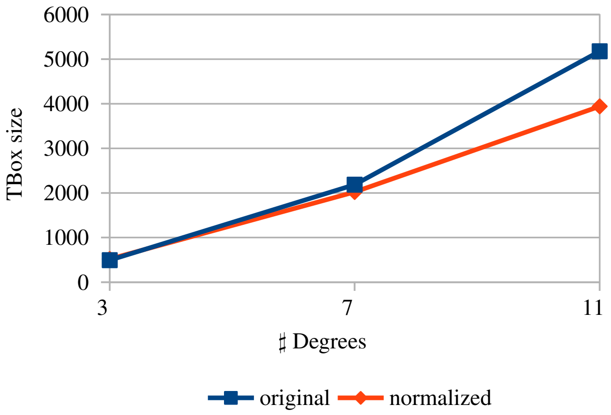

Fig. 2 The sizes of the crispified TBoxes (the number of occurrences of concept and role names) compared between the original LUBM TBox and its normalized variant

However, for only 3 degrees of membership, the avoided exponential blow-up does not outweigh the overhead of introducing auxiliary concept names for the normalization in practice.

Finally, it should be noted that the query answering time actually *decreases* when the number of truth degrees is increased (for both the normalized and the original TBox). This is not relevant for our experiments, but is simply a consequence of the Łukasiewicz semantics: When more fine-grained truth degrees are randomly assigned to all ABox assertions, the chance is larger that the degree to which a query atom is satisfied is reduced to 0, and hence does not yield an answer tuple (see Table 1). And with a decreased number of answers, the queries need less time to be answered.

5.3 Evaluating Conjunctive Query Answering

To analyze the practical usefulness of the reduction approach for CQ answering, we consider several CQs from the LUBM benchmark (namely #5, #7, and #9) over the normalized LUBM TBox.

An additional parameter that is evaluated here is the percentage of crisp concepts and roles appearing in the ontology; 0, 20, 80, or 100 % of all names are randomly chosen to be crisp.

Recall that a crisp concept or role name can only take the values 0 or 1. For such concepts and roles, the reduction process is more light-weight, since it results in a smaller number of concept names in the reduced ontology [2,5].

For each chain size, number of universities, and percentage of crisp symbols, we evaluated three *threshold CQs*[10] that extend the CQs in the LUBM benchmark.

[10] Note that for answering fuzzy CQs instead of threshold CQs, the runtime will increase by an exponential factor, depending on the number of atoms in the fuzzy CQ.

The evaluation was performed again w.r.t. the Łukasiewicz semantics. The results are depicted in Table 6.

We again see a counter-intuitive behavior, namely that the query answering time decreases with smaller percentage of crisp elements (and hence increased TBox size). The reason lies again with the Łukasiewicz semantics: queries over "more fuzzy" TBoxes will have less answers, and hence less effort is required to answer them. Nevertheless, we can see that the times required to answer threshold CQs are of the same order of magnitude as for classical CQs. Again the total runtime is dominated by the preprocessing time of PAGOdA, and hence mainly by the TBox size. This is good news, since the preprocessing step only needs to be performed once as long as the input TBox does not change.

Moreover, our approach is scalable for large ABoxes (as much as this is the case for a crisp ontology) since the crispification mainly affects the size of the TBox, but there is nearly no difference in size between a fuzzy ABox and its crispified version (see also Table 7).

As a final note, in practical applications of fuzzy ontologies one would hardly expect the ontology to have only fuzzy concept and role names, but rather that a smaller number of fuzzy names complements a large number of crisp names that describe precise knowledge. This is closer to our 80 % scenario, which does not differ much from the purely crisp case in terms of runtime.

6 Related Work

Non-fuzzy representations of fuzzy DLs have been extensively studied for several families of languages. The methods can be classified based on their fuzzy and DL expressivity used. Based on the Zadeh family of fuzzy logic operators, reduction techniques and optimizations have been examined for the fuzzy extensions of the $\mathcal{ALCH}$ [37], $\mathcal{SHOIN}$ [36], and $\mathcal{SROIQ}$ [3] languages, while an experimental evaluation of the reduction technique for the fuzzy version of $\mathcal{SHIN}$ is presented in [19]. For Gödel semantics, a reduction procedure for the DL $\mathcal{SROIQ}$ is considered in [7]. This procedure is extended in [8] for a language combining Gödel and Zadeh semantics. The reduction technique for the Łukasiewicz t-norm over $\mathcal{SROIQ}$ is studied in [10], and for arbitrary finite t-norms in [8,11]. It should be noted, as we show in Example 8, that the reduction of qualified number restrictions proposed in [8,10,28] is incorrect. This family of algorithms has been implemented in the DeLorean reasoner [5,6,9]. Based on a different approach, a family of fuzzy DLs using α-cuts as atomic concepts and roles is considered in [26].

Conjunctive query answering for fuzzy DLs has mostly been studied for the *DL-Lite* family of languages. In [38, 39], the problem of evaluating top-k queries in fuzzy *DL-*

Table 6 Time needed for preprocessing by PAGOdA (in s) and query execution time (in ms) for three queries and different percentages of crisp names

♯ Degrees	Crisp %	TBox size	♯ Universities											
			1				15				30			
			Preprocessing (in s)	Query 5 (in ms)	Query 7	Query 9	Preprocessing (in s)	Query 5 (in ms)	Query 7	Query 9	Preprocessing (in s)	Query 5 (in ms)	Query 7	Query 9
-	100	207	8.36	7.2	10.2	3.2	8.52	23.6	112.4	24.8	8.64	32.6	172.8	44.4
3	0	518	8.94	4.8	6.8	3.0	9.36	18.2	73.6	9.6	9.76	34.6	102.2	18.2
3	20	412	8.77	4.6	8.0	2.2	9.13	18.6	87.2	9.2	9.32	29.6	139.6	22.2
3	80	267	9.10	8.8	11.2	3.2	8.78	26.6	130.2	23.2	9.16	37.4	173.6	39.4
7	0	2022	11.88	5.4	3.0	2.4	12.49	17.2	14.2	3.6	13.05	21.8	29.0	6.4
7	20	1412	10.36	3.8	5.8	2.4	11.03	14.6	50.2	4.2	11.56	17.4	91.2	7.6
7	80	519	8.78	8.0	10.8	2.8	9.12	23.8	132.0	26.8	9.49	36.8	162.6	42.4
11	0	3942	18.63	2.2	3.4	2.6	19.52	10.6	11.4	3.6	20.17	13.2	17.6	4.4
11	20	2700	13.38	2.8	4.6	2.8	14.20	13.4	40.6	3.4	14.71	14.8	60.2	5.0
11	80	803	9.22	8.0	10.6	3.0	9.47	27.0	113.4	23.2	9.70	40.6	169.2	40.8

Lite is considered. In [32], the authors present a variety of query languages that can be used for querying fuzzy *DL-Lite* ontologies and adapt classical query rewriting techniques for answering these queries. A similar approach is taken in [30].

A tableaux algorithm for conjunctive query answering for fuzzy CARIN, a language combining the DL $\mathcal{ALCNR}$ with Horn rules under Zadeh semantics, is provided in [29]. Another algorithm for answering expressive fuzzy conjunctive queries is presented in [17,18]. The algorithm allows the occurrence of both lower bound and the upper bound of thresholds in a query atom over the DL $\mathcal{SHIN}$ extended with the Zadeh semantics. Finally, practical approaches for storing and querying fuzzy knowledge in the Semantic Web have been also investigated [35].

7 Conclusions and Future Work

This paper focuses on how a classical representation of ontologies written in finitely valued fuzzy DLs can be adopted to solve the threshold and fuzzy conjunctive query answering problems. These problems are reduced to equivalent (U)CQ answering problems in classical DLs. The correctness of the suggested technique is proved and its complexity is studied for different variants of $\mathscr{C}$-$\mathcal{SROIQ}$. As far as we know, no similar theoretical results have been presented. The proofs rely on the fact that each model of a fuzzy ontology $\mathcal{O}$ can be mapped to a model of its reduced crisp form $\mathcal{O}_\mathsf{c}$ and vice versa, thus showing the soundness and completeness of the reduction technique.

To verify the correctness of our approach, we have corrected and extended the proofs sketched in [10], providing the first full proof for this result in the literature. Additionally, we have improved the size of the reduction from fuzzy $\mathcal{SROIN}$ ontologies to classical $\mathcal{SROIN}$ ontologies by introducing a linear normalization step. After this step, the obtained classical ontology can be exponentially smaller than the one obtained by previous crispification approaches. The smaller size promises lower runtimes of the query answering procedure. We have also suggested a reduction of qualified number restrictions, which uses role disjunctions to express a partitioning of a role.

In our evaluation of a prototype implementation of this approach, based on the DeLorean system, we have demonstrated that the overhead involved in using a finitely valued fuzzzy description logic instead of a classical DL is reasonable. In a scenario where only the data and queries change frequently, while the TBox remains fixed, the normalization and the reduction of a finitely valued TBox can be computed "off-line". It is thus sufficient to consider the time required for query answering, which does not increase significantly when using more than two membership degrees.

Future work involves examining if available optimization techniques for fuzzy and classical DLs can be applied to improve the performance of these algorithms. On the theoretical side, we will investigate the possibility of extending the reduction to deal with qualified number restrictions without resorting to non-standard role constructors, obtaining a polynomial reduction for any finitely valued fuzzy description logic extending OWL 2 DL.

Acknowledgments This work was partially supported by the German Research Foundation (DFG) under the research Grant BA 1122/17-1 (FuzzyDL), the Collaborative Research Center 912 "Highly Adaptive Energy-Efficient Computing," and the Cluster of Excellence "Center for Advancing Electronics Dresden"; it was developed while R. Peñaloza was affiliated with TU Dresden and the Center for Advancing Electronics Dresden, Germany. We also want to thank Fernando Bobillo for providing us with a binary of the DeLorean system, and the anonymous reviewers for their valuable comments on earlier drafts of this paper.

Appendix

Proof of Theorem 12

Table 7 depicts the reduction rules for transforming a $\mathscr{C}$-$\mathcal{SROIN}$ ontology $\mathcal{O}$ into a classical $\mathcal{SROIN}$ ontology $\mathcal{O}_\mathsf{c}$. In this table, we use the notation

$$\ominus^-(d) := \max\{d' \in \mathscr{C} \mid \ominus d' \geqslant d\}.$$

Likewise, we define $\otimes^-(d)$ as the set of all pairs $(d_1, d_2) \in \mathscr{C}^2$ that satisfy $d_1 \otimes d_2 \geqslant d$ and are minimal w.r.t. the component-wise ordering on $\mathscr{C}^2$. This means that all elements of $\otimes^-(d)$ are incomparable, i.e., for all (d_1, d_2), $(d_1', d_2') \in \otimes^-(d)$ we have either $d_1 > d_1'$ and $d_2 < d_2'$ or vice versa. The set $\oplus^-(d)$ is defined analogously. For the implication, we need a slightly different definition, characterizing all pairs of elements whose implication does *not* exceed a specified value d. More precisely, we define $\Rightarrow^-(d)$ as the set of all $(d_1, d_2) \in \mathscr{C}^2$ satisfying $d_1 \Rightarrow d_2 < d$, and minimize here w.r.t. the first component and maximize w.r.t. the second component since $\Rightarrow$ is antitone in the first argument and monotone in the second argument.

Note that all expressions of the form $> d$ in Table 7 are well defined since we have $d < 1$ in all such cases. In particular, it holds that $\ominus^-(d) < 1$ whenever $d > 0$, and $d_2 < 1$ for all $(d_1, d_2) \in \Rightarrow^-(d)$. We can now prove Theorem 12. In the following, let $\mathcal{O}$ be an arbitrary (not necessarily normalized) ontology in $\mathscr{C}$-$\mathcal{SROIN}$, and $\mathcal{O}_\mathsf{c}$ be its reduced form according to Table 7.

Soundness

Consider first a classical model $\mathcal{J}$ of $\mathcal{O}_\mathsf{c}$. We construct a fuzzy interpretation $\mathcal{I}$, with the goal of showing that $\mathcal{I}$ is a

Table 7 Mapping of concepts, roles, and axioms to classical $\mathcal{SROIN}$

$\rho(\top, \geqslant d)$	$\top$
$\rho(\bot, \geqslant d)$	$\bot$
$\rho(A, \geqslant d)$	$A_{\geqslant d}$
$\rho(\neg C, \geqslant d)$	$\neg\rho(C, > \ominus^{-}(d))$
$\rho(C \sqcap D, \geqslant d)$	$\bigsqcup_{(d_1,d_2)\in\otimes^{-}(d)} \big(\rho(C, \geqslant d_1) \sqcap \rho(D, \geqslant d_2)\big)$
$\rho(C \sqcup D, \geqslant d)$	$\bigsqcup_{(d_1,d_2)\in\oplus^{-}(d)} \big(\rho(C, \geqslant d_1) \sqcap \rho(D, \geqslant d_2)\big)$
$\rho(\exists r.C, \geqslant d)$	$\bigsqcup_{(d_1,d_2)\in\otimes^{-}(d)} \exists\rho(r, \geqslant d_1).\rho(C, \geqslant d_2)$
$\rho(\forall r.C, \geqslant d)$	$\bigsqcap_{(d_1,d_2)\in\Rightarrow^{-}(d)} \forall\rho(r, \geqslant d_1).\rho(C, > d_2)$
$\rho(\{d_1/o_1, \ldots, d_m/o_m\}, \geqslant d)$	$\{o_i \mid d_i \geqslant d, i \in \{1, \ldots, m\}\}$
$\rho(\geq m\, r, \geqslant d)$	$\geq m\, \rho(r, \geqslant d)$
$\rho(\leq m\, r, \geqslant d)$	$\leq m\, \rho(r, > \ominus^{-}(d))$
$\rho(\exists r.\mathrm{Self}, \geqslant d)$	$\exists\rho(r, \geqslant d).\mathrm{Self}$
$\rho(r, \geqslant d)$	$r_{\geqslant d}$
$\rho(r^{-}, \geqslant d)$	$r^{-}_{\geqslant d}$
$\rho(u, \geqslant d)$	u
$\kappa(C(a) \geqslant d)$	$\rho(C, \geqslant d)(a)$
$\kappa(C(a) \leqslant d)$	$\neg\rho(C, > d)(a)$
$\kappa(r(a, b) \geqslant d)$	$\rho(r, \geqslant d)(a, b)$
$\kappa(r(a, b) \leqslant d)$	$\neg\rho(r, > d)(a, b)$
$\kappa(a \neq b)$	$a \neq b$
$\kappa(a = b)$	$a = b$
$\kappa(\langle C \sqsubseteq D \geqslant d\rangle)$	$\bigcup_{(d_1,d_2)\in\Rightarrow^{-}(d)} \big\{\rho(C, \geqslant d_1) \sqsubseteq \rho(D, > d_2)\big\}$
$\kappa(\langle r_1 r_2 \sqsubseteq r \geqslant d\rangle)$	$\bigcup_{(d_1,d')\in\Rightarrow^{-}(d),\ (d_2,d_3)\in\Rightarrow^{-}(d'_{\mathrm{next}})} \big\{\rho(r_1, \geqslant d_1)\rho(r_2, \geqslant d_2) \sqsubseteq \rho(r, > d_3)\big\}$
$\kappa(\langle r_1 \sqsubseteq r_2 \geqslant d\rangle)$	$\kappa(\langle r_1 u \sqsubseteq r_2 \geqslant d\rangle)$
$\kappa(\mathrm{trans}(r))$	$\kappa(rr \sqsubseteq r)$
$\kappa(\mathrm{dis}(r_1, r_2))$	$\mathrm{dis}(\rho(r_1, > 0), \rho(r_2, > 0))$
$\kappa(\mathrm{ref}(r))$	$\mathrm{ref}(\rho(r, \geqslant 1))$
$\kappa(\mathrm{irr}(r))$	$\mathrm{irr}(\rho(r, > 0))$
$\kappa(\mathrm{sym}(r))$	$\kappa(r \sqsubseteq r^{-})$
$\kappa(\mathrm{asy}(r))$	$\mathrm{asy}(\rho(r, > 0))$

model of $\mathcal{O}$, as follows (for all $x, y \in \Delta^{\mathcal{J}}$, $a \in \mathsf{N_I}$, $A \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, and $d \in \mathscr{C}$):

$$\Delta^{\mathcal{I}} := \Delta^{\mathcal{J}}$$
$$a^{\mathcal{I}} := a^{\mathcal{J}}$$
$$A^{\mathcal{I}}(x) := \max\left\{d \mid x \in A^{\mathcal{J}}_{\geqslant d}\right\}$$
$$r^{\mathcal{I}}(x, y) := \max\left\{d \mid (x, y) \in r^{\mathcal{J}}_{\geqslant d}\right\}$$

To show that $\mathcal{I}$ is a model of $\mathcal{O}$, we first prove the following proposition:

Proposition 19 *Let C be a concept, r a role, $x, y \in \Delta^{\mathcal{I}}$, and $d \in \mathscr{C}_{>}0$. Then, we have*

$$C^{\mathcal{I}}(x) \geqslant d \textit{ iff } x \in \rho(C, \geqslant d)^{\mathcal{J}} \textit{ and}$$
$$r^{\mathcal{I}}(x, y) \geqslant d \textit{ iff } (x, y) \in \rho(r, \geqslant d)^{\mathcal{J}}.$$

Proof For role names r, the claim holds by the construction of $r^{\mathcal{I}}$ and the fact that $r_{\geqslant d_{\mathrm{next}}} \sqsubseteq r_{\geqslant d}$ is contained in $\mathcal{O}_{\mathsf{c}}$ for every $d \in \mathscr{C}_{<1}$. For the universal role u, the equivalence trivially holds since we have that $(x, y) \in u^{\mathcal{J}}$ and $u^{\mathcal{I}}(x, y) = 1$. Finally, for inverse roles it is an immediate consequence of the facts that $(r^{-})^{\mathcal{I}}(x, y) = r^{\mathcal{I}}(y, x)$, and $(x, y) \in (r^{-}_{\geqslant d})^{\mathcal{J}}$ iff $(y, x) \in r^{\mathcal{J}}_{\geqslant d}$.

The rest of the proposition is proved by induction on the structure of C. For concept names A, the equivalence holds by the definition of $A^{\mathcal{I}}$ and the fact that $A_{\geqslant d_{\text{next}}} \sqsubseteq A_{\geqslant d}$ is in $\mathcal{O}_{\mathsf{c}}$ for every $d \in \mathscr{C}_{<1}$. For the induction step, we consider all possible concept constructors:

Top Concept: The proof for this case is an immediate consequence of the facts that $\top^{\mathcal{I}}(x) = 1$, $\rho(\top, \geqslant d) = \top$, and $\top^{\mathcal{J}} = \Delta^{\mathcal{J}} = \Delta^{\mathcal{I}}$.

Bottom Concept: For bottom, we have $\bot^{\mathcal{I}}(x) = 0$, $\rho(\bot, \geqslant d) = \bot$, and $\bot^{\mathcal{J}} = \emptyset$.

Concept Negation: We have that $x \in \rho(\neg C, \geqslant d)^{\mathcal{J}}$ holds if and only if $x \notin \rho(C, > \ominus^-(d))^{\mathcal{J}}$. By the induction hypothesis, this is equivalent to $C^{\mathcal{I}}(x) \leqslant \max\{d' \in \mathscr{C} \mid \ominus d' \geqslant d\}$. Since $\ominus$ is antitone, this is finally equivalent to $(\neg C)^{\mathcal{I}}(x) = \ominus C^{\mathcal{I}}(x) \geqslant d$.

Concept Conjunction: If $C_1^{\mathcal{I}}(x) \otimes C_2^{\mathcal{I}}(x) \geqslant d$, then by the definition of $\otimes^-(d)$ there is at least one pair $(d_1, d_2) \in \otimes^-(d)$ such that $C_1^{\mathcal{I}}(x) \geqslant d_1$ and $C_2^{\mathcal{I}}(x) \geqslant d_2$. Since $d > 0$, we also know that $d_1 > 0$ and $d_2 > 0$. Thus, by the induction hypothesis we have

$$x \in \big(\rho(C_1, \geqslant d_1) \sqcap \rho(C_2, \geqslant d_2)\big)^{\mathcal{J}} \subseteq \rho(C_1 \sqcap C_2, \geqslant d)^{\mathcal{J}}.$$

Conversely, suppose that $x \in (\rho(C_1,\geqslant d_1) \sqcap \rho(C_2,\geqslant d_2))^{\mathcal{J}}$ holds for some $(d_1,d_2) \in \otimes^-(d)$. Then, by induction hypothesis, $C_1^{\mathcal{I}}(x) \geqslant d_1$ and $C_2^{\mathcal{I}}(x) \geqslant d_2$. Because of the monotonicity of $\otimes$ we then get that $C_1^{\mathcal{I}}(x) \otimes C_2^{\mathcal{I}}(x) \geqslant d_1 \otimes d_2 \geqslant d$, as we wanted to show.

The proof for disjunction is similar to the proof for conjunction.

Existential Restriction: Suppose that $(\exists r.C)^{\mathcal{I}}(x) \geqslant d$. Since $\mathscr{C}$ is finite, there must exist some $y \in \Delta^{\mathcal{I}}$ with $r^{\mathcal{I}}(x,y) \otimes C^{\mathcal{I}}(y) \geqslant d$. We have $r^{\mathcal{I}}(x,y) \geqslant d_1$ and $C^{\mathcal{I}}(y) \geqslant d_2$ for some $(d_1,d_2) \in \otimes^-(d)$. By induction, we get $(x,y) \in \rho(r, \geqslant d_1)^{\mathcal{J}}$ and $y \in \rho(C, \geqslant d_2)^{\mathcal{J}}$. Therefore,

$$x \in \big(\exists \rho(r,\geqslant d_1).\rho(C,\geqslant d_2)\big)^{\mathcal{J}} \subseteq \rho(\exists r.C, \geqslant d)^{\mathcal{J}}.$$

Conversely, suppose that $x \in (\exists\rho(r,\geqslant d_1).\rho(C,\geqslant d_2))^{\mathcal{J}}$ for some pair $(d_1,d_2)\in\otimes^-(d)$. Thus, there exists y with $(x,y)\in\rho(r,\geqslant d_1)^{\mathcal{J}}$ and $y \in \rho(C,\geqslant d_2)^{\mathcal{J}}$. By induction, we have that $r^{\mathcal{I}}(x,y)\geqslant d_1$ and $C^{\mathcal{I}}(y)\geqslant d_2$, and therefore

$$(\exists r.C)^{\mathcal{I}}(x) \geqslant r^{\mathcal{I}}(x,y)\otimes C^{\mathcal{I}}(y) \geqslant d_1\otimes d_2 \geqslant d.$$

Universal Restriction: If $(\forall r.C)^{\mathcal{I}}(x) \geqslant d$, then for all $y \in \Delta^{\mathcal{I}}$ we have $\big(r^{\mathcal{I}}(x,y) \Rightarrow C^{\mathcal{I}}(y)\big) \geqslant d$. Consider any $y \in \Delta^{\mathcal{I}}$ and $(d_1,d_2) \in {\Rightarrow^-}(d)$ with $r^{\mathcal{I}}(x,y)\geqslant d_1$. If $C^{\mathcal{I}}(y) \leqslant d_2$, then we immediately get that

$$r^{\mathcal{I}}(x,y) \Rightarrow C^{\mathcal{I}}(y) \leqslant d_1 \Rightarrow d_2 < d,$$

which contradicts our assumption. Thus, we must have $C^{\mathcal{I}}(y) > d_2$, and hence $x \in \big(\forall\rho(r,\geqslant d_1).\rho(C,> d_2)\big)^{\mathcal{J}}$ by the induction hypothesis (as $d_1 > 0$). As this argument applies to all pairs $(d_1,d_2)\in{\Rightarrow^-}(d)$, we obtain $x\in\rho(\forall r.C,\geqslant d)^{\mathcal{J}}$, as required.

For the opposite direction, assume that $(\forall r.C)^{\mathcal{I}}(x) < d$. Then there must be a $y \in \Delta^{\mathcal{I}}$ such that $r^{\mathcal{I}}(x,y)\Rightarrow C^{\mathcal{I}}(y) < d$. By the definition of ${\Rightarrow^-}(d)$, we can find a pair $(d_1,d_2)\in{\Rightarrow^-}(d)$ with $r^{\mathcal{I}}(x,y)\geqslant d_1$ and $C^{\mathcal{I}}(y)\leqslant d_2$. Since $d_1>0$, the induction hypothesis yields that $(x,y)\in\rho(r,\geqslant d_1)^{\mathcal{J}}$ and $y\notin\rho(C,>d_2)^{\mathcal{J}}$. But then, this implies that $x\notin\rho(\forall r.C,\geqslant d)^{\mathcal{J}}$.

Nominals: Consider the case where $C=\{d_1/o_1,\dots,d_m/o_m\}$ such that $o_1,\dots,o_m \in \mathsf{N_I}$ and $d_1,\dots d_m \in \mathscr{C}_{>0}$. Then, $C^{\mathcal{I}}(x)\geqslant d$ iff $x = o_i^{\mathcal{I}}$ for some $i\in\{1,\dots,m\}$ with $d_i\geqslant d$, which in turn is equivalent to

$$x \in \{o_i^{\mathcal{I}} \mid d_i \geqslant d,\ i\in\{1,\dots,m\}\} = \rho(C,\geqslant d)^{\mathcal{J}}.$$

Unqualified Number Restrictions: The fact that $({\geq} m\, r)^{\mathcal{I}}(x)\geqslant d$ is equivalent to the existence of m different elements $y_1,\dots,y_m\in\Delta^{\mathcal{I}}$ such that $r^{\mathcal{I}}(x,y_i)\geqslant d$ holds for all $i\in\{1,\dots,m\}$. This is in turn equivalent to the existence of such y_i with $(x,y_i)\in\rho(r,\geqslant d)^{\mathcal{J}}$ for all i, and hence to $x\in({\geq} m\,\rho(r,\geqslant d))^{\mathcal{J}}$.

The proof for unqualified at-most restrictions can be obtained by a combination of previous arguments for ${\geq} m\, r$ and $\neg C$.

Local Reflexivity: We have $(\exists r.\mathrm{Self})^{\mathcal{I}}(x)\geqslant d$ iff $r^{\mathcal{I}}(x,x)\geqslant d$, which is equivalent to $(x,x)\in\rho(r,\geqslant d)^{\mathcal{J}}$, and to $x\in(\exists\rho(r,\geqslant d).\mathrm{Self})^{\mathcal{J}}$. □

To finish the proof of the first direction of Theorem 12, it remains to show the following lemma.

Lemma 20 *If $\mathcal{J}$ is a classical model of $\mathcal{O}_{\mathsf{c}}$, then $\mathcal{I}$ is a fuzzy model of $\mathcal{O}$.*

Proof We need to show that $\mathcal{I}$ satisfies all axioms in $\mathcal{O}$:

Concept Assertions: Suppose that $\mathcal{O}$ contains the concept assertion $C(a)\geqslant d$, and thus $\mathcal{O}_{\mathsf{c}}$ contains the $\rho(C,\geqslant d)(a)$. Since $\mathcal{J}$ is a model of $\mathcal{O}_{\mathsf{c}}$, we have $a^{\mathcal{J}}\in\rho(C,\geqslant d)^{\mathcal{J}}$, and by Proposition 19 that $C^{\mathcal{I}}(a^{\mathcal{I}})\geqslant d$. Similarly, for an assertion $C(a)\leqslant d$ in $\mathcal{O}$, we have $a^{\mathcal{J}}\notin\rho(C,>d)^{\mathcal{J}}$, and thus $C^{\mathcal{I}}(a^{\mathcal{I}})\leqslant d$.

Other ABox Axioms: The proof for role assertions can be obtained by adapting the proof for concept assertions. Axioms of the form $a\neq b$, $a=b$ are trivially satisfied since $\Delta^{\mathcal{I}}=\Delta^{\mathcal{J}}$ and for every individual $a\in\mathsf{N_I}$ we have $a^{\mathcal{I}}=a^{\mathcal{J}}$.

Concept Inclusions: Suppose that our ontology contains the concept inclusion $\langle C \sqsubseteq D \geqslant d\rangle$ and assume that there is a $x \in \Delta^{\mathcal{I}}$ such that $C^{\mathcal{I}}(x) \Rightarrow D^{\mathcal{I}}(x) < d$. Thus, there exists a pair $(d_1, d_2) \in \Rightarrow^{-}(d)$ such that $C^{\mathcal{I}}(x) \geqslant d_1$ and $D^{\mathcal{I}}(x) \leqslant d_2$. Since $d_1 > 0$, Proposition 19 yields $x \in \rho(C, \geqslant d_1)^{\mathcal{J}}$ and $x \notin \rho(D, > d_2)^{\mathcal{J}}$, which contradicts the fact that $\mathcal{J}$ satisfies $\rho(C, \geqslant d_1) \sqsubseteq \rho(D, > d_2)$.

Role Inclusions: Suppose that our ontology contains the role inclusion $\langle r_1 r_2 \sqsubseteq r \geqslant d\rangle$ and it holds that

$$\left(r_1^{\mathcal{I}}(x, y) \otimes r_2^{\mathcal{I}}(y, z)\right) \Rightarrow r^{\mathcal{I}}(x, z) < d,$$

or equivalently,

$$r_1^{\mathcal{I}}(x, y) \Rightarrow \left(r_2^{\mathcal{I}}(y, z) \Rightarrow r^{\mathcal{I}}(x, z)\right) < d,$$

for some $x, y, z \in \Delta^{\mathcal{I}}$. Then, there exist $(d_1, d') \in \Rightarrow^{-}(d)$ such that $r_1^{\mathcal{I}}(x, y) \geqslant d_1$ and $r_2^{\mathcal{I}}(y, z) \Rightarrow r^{\mathcal{I}}(x, z) \leqslant d'$. The latter implies the existence of $(d_2, d_3) \in \Rightarrow^{-}(d'_{\text{next}})$ with $r_2^{\mathcal{I}}(y, z) \geqslant d_2$ and $r^{\mathcal{I}}(x, z) \leqslant d_3$. Proposition 19 yields that $(x, y) \in \rho(r_1, \geqslant d_1)^{\mathcal{J}}$, $(y, z) \in \rho(r_2, \geqslant d_2)^{\mathcal{J}}$, and $(x, z) \notin \rho(r_3, > d_3)^{\mathcal{J}}$, which contradicts the fact that $\mathcal{J}$ satisfies $\rho(r_1, \geqslant d_1)\rho(r_2, \geqslant d_2) \sqsubseteq \rho(r, > d_3)$.

Disjoint Role Axioms: Suppose that our ontology contains the axiom $\text{dis}(r_1, r_2)$. We show that for all $x, y \in \Delta^{\mathcal{I}}$, either $r_1^{\mathcal{I}}(x, y) = 0$ or $r_2^{\mathcal{I}}(x, y) = 0$. Since $\mathcal{J}$ satisfies $\mathcal{O}_{\mathsf{c}}$, $\rho(r_1, > 0)^{\mathcal{J}} \cap \rho(r_2, > 0)^{\mathcal{J}} = \emptyset$. By Proposition 19, there can be no pair $x, y \in \Delta^{\mathcal{I}}$ such that $r_1^{\mathcal{I}}(x, y) > 0$ and $r_2^{\mathcal{I}}(x, y) > 0$, as we wanted to show.

The proofs for the other role axioms are similar. □

Completeness

Conversely, we consider a fuzzy model $\mathcal{I}$ of $\mathcal{O}$, and define the classical interpretation $\mathcal{J}$ as follows (for all $x, y \in \Delta^{\mathcal{I}}$, $a \in \mathsf{N_I}$, $A \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, and $d \in \mathscr{C}_{>0}$):

$$\begin{aligned}
\Delta^{\mathcal{J}} &:= \Delta^{\mathcal{I}}\\
a^{\mathcal{J}} &:= a^{\mathcal{I}}\\
A_{\geqslant d}^{\mathcal{J}} &:= \{x \mid A^{\mathcal{I}}(x) \geqslant d\}\\
r_{\geqslant d}^{\mathcal{J}} &:= \{(x, y) \mid r^{\mathcal{I}}(x, y) \geqslant d\}
\end{aligned}$$

We again prove a connection similar to the one of Proposition 19.

Proposition 21 *Let C be a concept, r a role, $x, y \in \Delta^{\mathcal{I}}$, and $d \in \mathscr{C}_{>}0$. Then, we have*

$$x \in \rho(C, \geqslant d)^{\mathcal{J}} \text{ iff } C^{\mathcal{I}}(x) \geqslant d \text{ and}$$
$$(x, y) \in \rho(r, \geqslant d)^{\mathcal{J}} \text{ iff } r^{\mathcal{I}}(x, y) \geqslant d.$$

Proof The proof is nearly the same as for Proposition 19, the only difference being the induction base cases. But it is easy to show the claim for concept and role names, given the definition of $\mathcal{J}$. □

Lemma 22 *If $\mathcal{I}$ is a fuzzy model of $\mathcal{O}$, then $\mathcal{J}$ is a classical model of $\mathcal{O}_{\mathsf{c}}$.*

Proof We need to show that $\mathcal{J}$ satisfies all axioms in $\mathcal{O}_{\mathsf{c}}$:

Concept Assertions: Suppose that $\mathcal{O}_{\mathsf{c}}$ contains the concept assertion $\rho(C, \geqslant d)(a)$. By the construction of $\mathcal{O}_{\mathsf{c}}$, $C(a) \geqslant d$ appears in $\mathcal{O}$. Since $\mathcal{I}$ is a model of $\mathcal{O}$, we have $C^{\mathcal{I}}(a) \geqslant d$, and by Proposition 21 we get $a^{\mathcal{J}} \in \rho(C, \geqslant d)^{\mathcal{J}}$, as we wanted to show.

If $\mathcal{O}_{\mathsf{c}}$ contains an assertion $\neg\rho(C, > d)(a)$, then $C(a) \leqslant d$ appears in $\mathcal{O}$ and consequently $C^{\mathcal{I}}(a) \leqslant d$. By Proposition 21, we have that $a \notin \rho(C, > d)^{\mathcal{J}}$, as we wanted to show.

Other ABox Axioms: The proof for role assertions can be obtained by adapting the proof for concept assertions. Axioms of the form $a \neq b$, $a = b$ are trivially satisfied since $\Delta^{\mathcal{I}} = \Delta^{\mathcal{J}}$ and for every individual $a \in \mathsf{N_I}$ we have $a^{\mathcal{I}} = a^{\mathcal{J}}$.

Concept Inclusions: Suppose that our ontology contains a concept inclusion $\rho(C, \geqslant d_1) \sqsubseteq \rho(D, > d_2)$ that is not satisfied. Thus, there exists some $x \in \Delta^{\mathcal{J}}$ such that $x \in \rho(C, \geqslant d_1)^{\mathcal{J}}$ and $x \notin \rho(D, > d_2)^{\mathcal{J}}$ and, by Proposition 21, $C^{\mathcal{I}}(x) \geqslant d_1$ and $D^{\mathcal{I}}(x) \leqslant d_2$. By the construction of $\mathcal{O}_{\mathsf{c}}$, we have $\langle C \sqsubseteq D \geqslant d\rangle$ in $\mathcal{O}$ and $(d_1 \Rightarrow d_2) < d$ for some $d \in \mathscr{C}_{>0}$. By the properties of $\Rightarrow$, we get $C^{\mathcal{I}}(x) \Rightarrow D^{\mathcal{I}}(x) \leqslant d_1 \Rightarrow d_2 < d$, which contradicts our assumption that $\mathcal{I}$ is a model of $\mathcal{O}$.

All concept inclusions of the form $A_{\geqslant d_{\text{next}}} \sqsubseteq A_{\geqslant d}$ are trivially satisfied by the construction of $\mathcal{J}$.

Role Inclusions: Assume that a role inclusion

$$\rho(r_1, \geqslant d_1)\rho(r_2, \geqslant d_2) \sqsubseteq \rho(r, > d_3) \in \mathcal{O}_{\mathsf{c}}$$

is violated, i.e., there are three elements $x, y, z \in \Delta^{\mathcal{J}}$ such that we have $(x, y) \in \rho(r_1, \geqslant d_1)^{\mathcal{J}}$, $(y, z) \in \rho(r_2, \geqslant d_2)^{\mathcal{J}}$, and $(x, z) \notin \rho(r, > d_3)^{\mathcal{J}}$. Proposition 21 implies that:

$$r_1^{\mathcal{I}}(x, y) \geqslant d_1, \qquad r_2^{\mathcal{I}}(y, z) \geqslant d_2, \qquad r(x, z) \leqslant d_3. \tag{8}$$

By construction of $\mathcal{O}_{\mathsf{c}}$, we have that $\langle r_1 r_2 \sqsubseteq r \geqslant d\rangle \in \mathcal{O}$, $(d_1 \Rightarrow d') < d$, and $(d_2 \Rightarrow d_3) < d'_{\text{next}}$ for some $d, d' \in \mathscr{C}_{>0}$. We obtain

$$\left(d_1 \Rightarrow (d_2 \Rightarrow d_3)\right) \leqslant (d_1 \Rightarrow d') < d,$$

and hence $\left((d_1 \otimes d_2) \Rightarrow d_3\right) < d$.

Along with (8) and the monotonicity and antitonicity properties of the operators $\otimes$ and $\Rightarrow$, this implies that

$$\Big(\big(r_1^{\mathcal{I}}(x, y) \otimes r_2^{\mathcal{I}}(y, z)\big) \Rightarrow r^{\mathcal{I}}(x, z)\Big) < d$$

which is absurd since $\mathcal{I}$ is a model of $\mathcal{O}$ and $\langle r_1 r_2 \sqsubseteq r \geqslant d\rangle \in \mathcal{O}$.

All role inclusions of the form $r_{\geqslant d_{\text{next}}} \sqsubseteq r_{\geqslant d}$ are trivially satisfied by the construction of $\mathcal{J}$.

Disjoint Role Axioms: Suppose that $\mathcal{O}_{\mathsf{c}}$ contains the disjoint role axiom $\text{dis}(\rho(r_1, > 0), \rho(r_2, > 0))$. By construction of $\mathcal{O}_{\mathsf{c}}$, we also have that $\text{dis}(r_1, r_2) \in \mathcal{O}$, and therefore either $r_1(x, y) = 0$ or $r_2(x, y) = 0$ for all $x, y \in \Delta^{\mathcal{I}}$. Proposition 21 now implies that the axiom

$$\text{dis}(\rho(r_1, > 0), \rho(r_2, > 0))$$

is satisfied.

The proofs for the other role axioms are similar. □

Proof of Lemma 13

To determine the size of $\mathcal{O}_{\mathsf{c}}$ for a normalized $\mathscr{C}$-$\mathcal{SROIN}$ ontology $\mathcal{O}$, we start by analyzing the size of the sets $\otimes^{-}(d)$, $\oplus^{-}(d)$ and $\Rightarrow^{-}(d)$ (defined in the beginning of the proof of Theorem 12). It is clear that for every $d_1 \in \mathscr{C}$ there can be at most one element $d_2 \in \mathscr{C}$ such that (d_1, d_2) is contained in any of these sets. This is due to the minimization conditions in their definitions. Thus, the size of these sets is at most linear in the size of $\mathscr{C}$. Consequently, the size of any expression of the form $\rho(C, \geqslant d)$, where C is a complex concept that contains only one concept constructor, is at most linear in the sizes of C and $\mathscr{C}$ (cf. Table 7).

Since $\mathcal{O}$ is normalized, ABox axioms contain no complex concepts, and hence the size of $\kappa(\alpha)$ for any such axiom α is the same as the size of α.

Consider now a GCI $\alpha := \langle C \sqsubseteq D \geqslant d\rangle$ and its reduced form, containing a GCI $\rho(C, \geqslant d_1) \sqsubseteq \rho(D, > d_2)$ for each pair $(d_1, d_2) \in \Rightarrow^{-}(d)$. Since α contains at most one concept constructor, the size of each reduced axiom is linear in the sizes of α and $\mathscr{C}$. Moreover, there are at most linearly many such axioms (in the size of $\mathscr{C}$), bringing the total size of $\kappa(\alpha)$ to at most linear in the size of α and quadratic in the size of $\mathscr{C}$.

Likewise, for a role inclusion $\alpha := \langle r_1 r_2 \sqsubseteq r \geqslant d\rangle$ the number of pairs $(d_1, d') \in \Rightarrow^{-}(d)$ is linear in the size of $\mathscr{C}$, and for each of these pairs we additionally have to consider linearly many pairs of the form $(d_2, d_3) \in \Rightarrow^{-}(d'_{\text{next}})$. Thus, the same bounds are valid for role inclusions. The proof for the remaining role axioms is trivial.

In summary, the total size of $\mathcal{O}_{\mathsf{c}}$ is bounded linearly in the size of $\mathcal{O}$ and quadratically in the size of $\mathscr{C}$. □

References

1. Baader F, Borgwardt S, Peñaloza R (2015) On the decidability status of fuzzy $\mathcal{ALC}$ with general concept inclusions. Journal of Philosophical Logic 44(2):117–146. doi:10.1007/s10992-014-9329-3
2. Bobillo F (2015) The impact of crispness in fuzzy tractable ontologies: The case of fuzzy OWL 2 EL. IEEE Transactions on Fuzzy Systems doi:10.1109/TFUZZ.2015.2505329. In press
3. Bobillo F, Delgado M, Gómez-Romero J (2008) Optimizing the crisp representation of the fuzzy Description Logic $\mathcal{SROIQ}$. In: Uncertainty Reasoning for the Semantic Web I, pp. 189–206. Springer
4. Bobillo F, Delgado M, Gómez-Romero J (2009) Crisp representations and reasoning for fuzzy ontologies. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 17(4):501–530. doi:10.1142/S0218488509006121
5. Bobillo F, Delgado M, Gómez-Romero J (2012) DeLorean: A reasoner for fuzzy OWL 2. Expert Systems with Applications 39(1):258–272. doi:10.1016/j.eswa.2011.07.016
6. Bobillo F, Delgado M, Gómez-Romero J (2013) Reasoning in fuzzy OWL 2 with DeLorean. In: Bobillo F, da Costa PCG, d'Amato C, Fanizzi N, Laskey K, Laskey K, Lukasiewicz T, Nickles M, Pool M (eds.) Uncertainty Reasoning for the Semantic Web II, Lecture Notes in Computer Science, vol. 7123, pp. 119–138. Springer doi:10.1007/978-3-642-35975-0-7
7. Bobillo F, Delgado M, Gómez-Romero J, Straccia U (2009) Fuzzy Description Logics under Gödel semantics. International Journal of Approximate Reasoning 50(3):494–514
8. Bobillo F, Delgado M, Gómez-Romero J, Straccia U (2012) Joining Gödel and Zadeh fuzzy logics in fuzzy description logics. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 20(4):475–508. doi:10.1142/S0218488512500249
9. Bobillo F, Delgado M, Sanchez-Sanchez JC (2013) Parallel algorithms for fuzzy ontology reasoning. IEEE Transactions on Fuzzy Systems 21(4):775–781. doi:10.1109/TFUZZ.2012.2230266
10. Bobillo F, Straccia U (2011) Reasoning with the finitely many-valued Łukasiewicz fuzzy Description Logic $\mathcal{SROIQ}$. Information Sciences 181(4):758–778
11. Bobillo F, Straccia U (2013) Finite fuzzy Description Logics and crisp representations. In: Uncertainty Reasoning for the Semantic Web II, pp. 99–118. Springer
12. Borgwardt S, Distel F, Peñaloza R (2015) The limits of decidability in fuzzy description logics with general concept inclusions. Artificial Intelligence 218:23–55. doi:10.1016/j.artint.2014.09.001
13. Borgwardt S, Peñaloza R (2013) The complexity of lattice-based fuzzy description logics. Journal on Data Semantics 2(1):1–19
14. Borgwardt S, Peñaloza R (2014) Consistency reasoning in lattice-based fuzzy description logics. International Journal of Approximate Reasoning 55(9):1917–1938. doi:10.1016/j.ijar.2013.07.006
15. Calvanese D, De Giacomo G, Lenzerini M (1998) On the decidability of query containment under constraints. In: Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems, pp. 149–158. ACM
16. Calvanese D, Eiter T, Ortiz, M (2009) Regular path queries in expressive description logics with nominals. In: Boutilier C (ed.) Proc. of the 21st Int. Joint Conf. on Artificial Intelligence (IJCAI'09), pp. 714–720. AAAI Press. http://ijcai.org/papers09/Papers/IJCAI09-124.pdf
17. Cheng J, Ma Z, Zhang F, Wang X (2009) Deciding query entailment for fuzzy $\mathcal{SHIN}$ ontologies. In: Proceedings of the 4th Asian Conference on The Semantic Web, ASWC '09, pp. 120–134. Springer
18. Cheng J, Ma Z, Zhang F, Wang X (2009) Deciding query entailment in fuzzy Description Logic knowledge bases. In: Database and Expert Systems Applications, pp. 830–837. Springer

19. Cimiano P, Haase P, Ji Q, Mailis T, Stamou G, Stoilos G, Tran DT, Tzouvaras V (2008) Reasoning with large A-Boxes in fuzzy Description Logics using DL reasoners: an experimental evaluation. In: Proceedings of the ESWC Workshop on Advancing Reasoning on the Web: Scalability and Commonsense
20. Eiter T, Lutz C, Ortiz M, Šimkus M (2009) Query answering in description logics with transitive roles. In: Boutilier C (ed.) Proc. of the 21st Int. Joint Conf. on Artificial Intelligence (IJCAI'09), pp. 759–764. AAAI Press . http://ijcai.org/papers09/Papers/IJCAI09-131.pdf
21. Glimm B, Horrocks I, Motik B, Stoilos G, Wang Z (2014) HermiT: An OWL 2 reasoner. Journal of Automated Reasoning 53(3):245–269
22. Glimm B, Horrocks I, Sattler U (2008) Unions of conjunctive queries in $\mathcal{SHOQ}$. In: Brewka G, Lang J (eds.) Proc. of the 11th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'08), pp. 252–262. AAAI Press. http://www.aaai.org/Library/KR/2008/kr08-025.php
23. Guo Y, Pan Z, Heflin J (2005) LUBM: A benchmark for OWL knowledge base systems. Web Semantics: Science, Services and Agents on the World Wide Web 3(2):158–182
24. Gutiérrez-Basulto V, Ibañez-García Y, Kontchakov R, Kostylev EV (2013) Conjunctive queries with negation over DL-Lite: A closer look. In: Faber W, Lembo D (eds.) Proc. of the 7th Int. Conf. on Web Reasoning and Rule Systems (RR'13), Lecture Notes in Computer Science, vol. 7994, pp. 109–122. Springer-Verlag doi:10.1007/978-3-642-39666-3_9
25. Horrocks I, Kutz O, Sattler U (2006) The even more irresistible $\mathcal{SROIQ}$. In: Proc. of the 10th International Conference of Knowledge Representation and Reasoning (KR-2006), 6, pp. 57–67
26. Li Y, Xu B, Lu J, Kang D, Wang P (2005) A family of extended fuzzy Description Logics. In: Computer Software and Applications Conference, vol. 1, pp. 221–226. IEEE
27. Lutz C (2008) The complexity of conjunctive query answering in expressive description logics. In: Proc. of the 4th Int. Joint Conf. on Automated Reasoning (IJCAR'08), Lecture Notes in Artificial Intelligence, vol. 5195, pp. 179–193. Springer-Verlag doi:10.1007/978-3-540-71070-7_16
28. Mailis T, Peñaloza R, Turhan AY (2014) Conjunctive query answering in finitely-valued fuzzy description logics. In: Kontchakov R, Mugnier ML (eds.) Proceedings of the 8th International Conference on Web Reasoning and Rule Systems (RR 2014), vol. 8741, pp. 124–139. Springer
29. Mailis T, Stoilos G, Stamou G (2010) Expressive reasoning with Horn rules and fuzzy Description Logics. Knowledge and Information Systems 25(1):105–136
30. Straccia, U.: Foundations of Fuzzy Logic and Semantic Web Languages. CRC Press, UK (2013)
31. Möller R, Neuenstadt C, Özçep ÖL, Wandelt S (2013) Advances in accessing big data with expressive ontologies. In: Timm IJ, Thimm M (eds.) KI 2013: Advances in Artificial Intelligence - 36th Annual German Conference on AI, Koblenz, Germany, September 16-20, 2013. Proceedings, *Lecture Notes in Computer Science*, vol. 8077, pp. 118–129. Springer (2013)
32. Pan, J.Z., Stamou, G.B., Stoilos, G., Thomas, E.: Expressive querying over fuzzy DL-Lite ontologies. In: 20th International Workshop on Description Logics (2007)
33. Rosati R (2007) The limits of querying ontologies. In: Schwentick T, Suciu D (eds.) Proc. of the 11th Int. Conf. on Database Theory (ICDT'07), Lecture Notes in Computer Science, vol. 4353, pp. 164–178. Springer-Verlag (2007). doi:10.1007/11965893_12
34. Rudolph S, Krötzsch M, Hitzler P (2008) Cheap Boolean role constructors for description logics. In: Hölldobler S, Lutz C, Wansing H (eds.) Proceedings of the 11th European Conference on Logics in Artificial Intelligence (JELIA'08), Lecture Notes in Computer Science, vol. 5293, pp. 362–374. Springer doi:10.1007/978-3-540-87803-2_30
35. Simou N, Stoilos G, Tzouvaras V, Stamou GB, Kollias SD (2008) Storing and querying fuzzy knowledge in the Semantic Web. In: 7th International Workshop on Uncertainty Reasoning For the Semantic Web. Karlsruhe, Germany
36. Stoilos G, Stamou GB (2007) Extending fuzzy description logics for the semantic web. In: Golbreich C, Kalyanpur A, Parsia B (eds.) Proc. of the 3rd Int. Workshop on OWL: Experiences and Directions (OWLED'07), *CEUR Workshop Proceedings*, vol. 258 http://ceur-ws.org/Vol-258/paper12.pdf
37. Straccia U (2004) Transforming fuzzy description logics into classical description logics. In: Proc. of the 9th Eur. Conf. on Logics in Artificial Intelligence (JELIA'04), pp. 385–399. Springer
38. Straccia U (2006) Answering vague queries in fuzzy DL-Lite. In: Proceedings of the 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems,(IPMU-06), pp. 2238–2245
39. Straccia U (2006) Towards top-k query answering in Description Logics: the case of DL-Lite. In: Logics in Artificial Intelligence, Lecture Notes in Computer Science, pp. 439–451. Springer
40. Straccia U (2013) Foundations of Fuzzy Logic and Semantic Web Languages. CRC Press, USA
41. Straccia U (2014) On the top-k retrieval problem for ontology-based access to databases. In: Flexible Approaches in Data, Information and Knowledge Management, pp. 95–114. Springer
42. Zhou Y, Grau BC, Nenov Y, Horrocks I (2015) PAGOdA: Pay-as-you-go ABox reasoning. In: Calvanese D, Konev B (eds.) Proceedings of the 28th International Workshop on Description Logics., CEUR Workshop Proceedings, vol. 1350. CEUR-WS.org
43. Zhou Y, Nenov Y, Grau BC, Horrocks I (2014) Pay-as-you-go OWL query answering using a triple store. In: Proc. of the 28th AAAI Conf. on Artificial Intelligence (AAAI'14), pp. 1142–1148. AAAI Press (2014). http://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8232