Deciding Fuzzy Description Logics by Type Elimination*

Uwe Keller[1] and Stijn Heymans[2]

[1] Semantic Technology Institute (STI) Innsbruck, University of Innsbruck, Austria. eMail: uwe.keller@sti2.at

[2] Knowledge-based Systems Group, Institute of Information Systems, Vienna University of Technology, Austria. eMail: heymans@kr.tuwien.ac.at

Abstract. We present a novel procedure $\mathbf{FixIt}(\mathbb{ALC})$ for deciding knowledge base satisfiability in the Fuzzy Description Logic (FDL) $\mathbb{ALC}$. $\mathbf{FixIt}(\mathbb{ALC})$ does not search for tree-structured models as in tableau-based proof procedures, but embodies a fixpoint-computation of canonical models that are not necessarily tree-structured. Conceptually, the procedure is based on a type-elimination process. Soundness, completeness and termination are proven. To the best of our knowledge it is the first *fixpoint-based* decision procedure for FDLs, hence introducing a new class of inference procedures into FDL reasoning.

1 Introduction

Description Logics (DLs) [1] are a popular family of formally well-founded and decidable knowledge representation languages. DLs have a wide range of applications, e.g., they form the basis for Semantic Web (SW) ontology languages used such as OWL [7]. Fuzzy Description Logics (FDLs) [13] extend DLs to represent *vague* concepts and relations, and as such are very well suited to cover for representing and reasoning with uncertainty, a requirement that naturally arises in many practical applications of knowledge-based systems, in particular the SW.

So far, reasoning in Fuzzy DLs is mainly based on tableau-methods (e.g. [13,12,4,11,16,3]). Further, [14] demonstrates how to use inference procedures for classical DLs to perform reasoning in (some) FDLs. Still, reasoning in FDLs is at least as hard as reasoning in classical (crisp) DLs. Even in DLs of modest expressivity (e.g. $\mathbb{ALC}$ [13,14,12] the fuzzy variant of the DL $\mathcal{ALC}$ [10]) the worst-case complexity of reasoning is significant even in restricted cases [13]. Therefore, it is clear that there can not be a *single* inference method that works well on *all* problems.

Consequently, our goal is to enrich the range of available methods for reasoning with FDLs with a fundamentally different approach. In practical applications of DLs (and hence FDLs) a particularly important feature for representing domain models is the support of so-called *general terminologies* (see e.g. [12]), i.e., the possibility to capture (potentially recursive) interdependencies between complex concepts in a domain

* This work has been partially funded by the European Commission under the LarKC project (FP7 - 215535). Stijn Heymans is supported by the Austrian Science Fund (FWF) under projects P20305 and P20840.

model. However, besides the tableau-based methods for DLs (e.g [12,4,16,3]) there are at present no other FDL inference methods which can deal with general terminologies. We want to provide an alternative to tableau-based methods that can deal with general terminologies.

The main contributions of the paper are as follows:

- We present a novel procedure $\mathbf{FixIt}(\mathbb{ALC})$ (cf. Section 3) for deciding knowledge base (KB) satisfiability in the FDL $\mathbb{ALC}$ (cf. Section 2).
- We formally prove soundness, completeness and termination of the algorithm (cf. Section 3).
- $\mathbf{FixIt}(\mathbb{ALC})$ generalizes a type-elimination-based decision procedure [8] for the (classical) modal logic $\mathbf{K}$ (i.e. $\mathcal{KBDD}$ [6]) to the FDL $\mathbb{ALC}$. Additionally we integrate (fuzzy) ABoxes and general TBoxes which are not dealt with in $\mathcal{KBDD}$.
- To the best of our knowledge it is the first *fixpoint-based* decision procedure that has been proposed for FDL introducing a *new class of inference procedures* into FDL reasoning.
- Besides the tableau-based methods in [12,4,16], it is the only approach to integrate general terminologies in FDL reasoning and the first non-tableau-based one that we are aware of. General terminologies are handled in a fundamentally different way than in standard tableau-based method such as [12,4].

Our method is interesting especially regarding the last aspect since the handling of general terminologies in standard tableau-based methods (e.g. [12,4]) is a *major* source of non-determinism and thus computational inefficiency. In our case no non-deterministic choice is introduced by terminologies.

2 Preliminaries

We introduce $\mathbb{ALC}$ [13], the fuzzy variant of the Description Logic $\mathcal{ALC}$ [10] (the latter can be seen as a syntactic variant of the multi-modal logic $\mathbf{K}_{(\mathbf{m})}$ [9]). $\mathbb{ALC}$ provides the starting point for more expressive FDLs [15] that have been proposed to fuzzify major fragments of OWL [7].

Syntax. Concept expressions are constructed from a signature $\Sigma = (\mathbf{C}, \mathbf{R}, \mathbf{I})$ with concept names $\mathbf{C}$, role names $\mathbf{R}$, and individual names $\mathbf{I}$. The set of concept expressions $\mathcal{C}(\Sigma)$ over Σ is defined as the smallest set of expressions that contains $\mathbf{C}$, $\top$ and is closed under the application of the concept constructors $C \sqcap D$ (intersection), $C \sqcup D$ (union), $\neg C$ (complement), and $\forall R.C$ (universal role restriction) for $R \in \mathbf{R}$ and $C, D \in \mathcal{C}(\Sigma)$. We allow expressions $\exists R.C$ for $C \in \mathcal{C}(\Sigma), R \in \mathbf{R}$ and $\bot$ and treat them as shortcuts for $\neg\forall R.\neg C$ and $\neg\top$ respectively. A TBox axiom (or general concept inclusion axiom (GCI)) is an expression of the form $C \sqsubseteq D$ s.t. $C, D \in \mathcal{C}(\Sigma)$. A terminology (or TBox) $\mathcal{T}$ is a finite set of TBox axioms. Syntactically, the vagueness of descriptions becomes explicit only when describing specific instances and their interrelations: a (fuzzy) ABox axiom is either a $\langle i : C \bowtie d\rangle$ or a $\langle R(i, i') \geq d\rangle$ s.t. $i, i' \in \mathbf{I}$, $d \in [0, 1]$, and $\bowtie \in \{\leq, \geq, =\}$. An ABox $\mathcal{A}$ is a finite set of ABox axioms. Finally, a knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ consists of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$. Let $Ind_{\mathcal{A}} \subseteq \mathbf{I}$ denote the individual names that occur in $\mathcal{A}$. We denote the set of all concept expressions that occur as subexpressions in $\mathcal{K}$ by $\mathsf{sub}(\mathcal{K})$.

Semantics. Semantically, vagueness is reflected in the use of fuzzy sets and relations when interpreting concepts and roles: an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a non-empty set $\Delta^{\mathcal{I}}$ called the domain, and a function $\cdot^{\mathcal{I}}$ which maps each concept name $C \in \mathbf{C}$ to a fuzzy set $C^{\mathcal{I}} : \Delta^{\mathcal{I}} \to [0,1]$, each role name $R \in \mathbf{R}$ to a fuzzy relation $R^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \to [0,1]$ and each individual name $i \in \mathbf{I}$ to an element $i^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. The interpretation function $\cdot^{\mathcal{I}}$ is extended to arbitrary concept expressions $C \in \mathcal{C}(\Sigma)$ as follows:

- $(C \sqcap D)^{\mathcal{I}}(o) = min(C^{\mathcal{I}}(o), D^{\mathcal{I}}(o))$
- $(C \sqcup D)^{\mathcal{I}}(o) = max(C^{\mathcal{I}}(o), D^{\mathcal{I}}(o))$
- $(\neg C)^{\mathcal{I}}(o) = 1 - C^{\mathcal{I}}(o)$
- $(\forall R.C)^{\mathcal{I}}(o) = inf_{o' \in \Delta^{\mathcal{I}}}\{max(1 - R^{\mathcal{I}}(o, o'), C^{\mathcal{I}}(o'))\}$
- $\top^{\mathcal{I}}(o) = 1$

for all $o \in \Delta^{\mathcal{I}}, C, D \in \mathcal{C}(\Sigma)$, $R \in \mathbf{R}$.

An interpretation $\mathcal{I}$ satisfies a TBox axiom $\alpha = C \sqsubseteq D$ iff. for all $o \in \Delta^{\mathcal{I}}$ it holds that $C^{\mathcal{I}}(o) \leq D^{\mathcal{I}}(o)$, i.e. C is a fuzzy subset of D. $\mathcal{I}$ satisfies an ABox axiom $\alpha = \langle i : C \bowtie d \rangle$ iff. $C^{\mathcal{I}}(i^{\mathcal{I}}) \bowtie d$. $\mathcal{I}$ satisfies an ABox axiom $\alpha = \langle R(i, i') \geq d \rangle$ iff. $R^{\mathcal{I}}(i^{\mathcal{I}}, i'^{\mathcal{I}}) \geq d$. In all these cases, we write $\mathcal{I} \models \alpha$. $\mathcal{I}$ satisfies a TBox $\mathcal{T}$ (or is a model of $\mathcal{T}$) iff. $\mathcal{I} \models \alpha$ for all $\alpha \in \mathcal{T}$. $\mathcal{I}$ satisfies an ABox $\mathcal{A}$ (or is a model of $\mathcal{A}$) iff. $\mathcal{I} \models \alpha$ for all $\alpha \in \mathcal{A}$. Finally, $\mathcal{I}$ satisfies a knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ (or is a model of $\mathcal{K}$) iff. $\mathcal{I} \models \mathcal{T}$ and $\mathcal{I} \models \mathcal{A}$.

Reasoning in $\mathbb{ALC}$. Given a fuzzy KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, fuzzy ABox axioms or GCIs α and concept expressions $C, D \in \mathcal{C}(\Sigma)$, we can analyze particular semantic characteristics and interdependencies: We say that $\mathcal{K}$ is *satisfiable* (or consistent) iff. there is a model $\mathcal{I}$ for $\mathcal{K}$. $\mathcal{K}$ *entails* α (denoted as $\mathcal{K} \models \alpha$) iff. all models $\mathcal{I}$ of $\mathcal{K}$ satisfy α. Concept C is subsumed by concept D (wrt. a KB $\mathcal{K}$) iff. $\mathcal{K} \models C \sqsubseteq D$. Two concepts C and D are called *equivalent* (wrt. a KB $\mathcal{K}$) iff. for any model $\mathcal{I}$ of $\mathcal{K}$ it holds that $C^{\mathcal{I}}(o) = D^{\mathcal{I}}(o)$ for all $o \in \Delta^{\mathcal{I}}$. Two concepts C and D are called *disjoint* (wrt. a KB $\mathcal{K}$) iff. for any model $\mathcal{I}$ of $\mathcal{K}$ it holds that there does not exists an $o \in \Delta^{\mathcal{I}}$ such that $C^{\mathcal{I}}(o) > 0$ and $D^{\mathcal{I}}(o) > 0$. A concept C is called satisfiable (wrt. a KB $\mathcal{K}$) iff. there exists a model $\mathcal{I}$ of $\mathcal{T}$ such that $C^{\mathcal{I}}(o) > 0$ for some $o \in \Delta^{\mathcal{I}}$. Further, one might want to compute the truth value bounds for a given ABox assertion α wrt. $\mathcal{K}$ to determine the possibility interval that is enforced for α by the background knowledge in $\mathcal{K}$: The *greatest lower bound* of α wrt. $\mathcal{K}$ is defined as $glb(\alpha, \mathcal{K}) := sup\{d \in [0,1] \mid \mathcal{K} \models \langle \alpha \geq d \rangle\}$ and the *least upper bound* of α wrt. $\mathcal{K}$ is defined as $lub(\alpha, \mathcal{K}) := inf\{d \in [0,1] \mid \mathcal{K} \models \langle \alpha \leq d \rangle\}$ (where $sup\, \emptyset = inf\, \emptyset = 0$). Computing $glb(\alpha, \mathcal{K})$ and $lub(\alpha, \mathcal{K})$ is usually called the *best truth value bounds* (BTVB) problem.

One of the most fundamental reasoning problems is to determine whether a given fuzzy KB $\mathcal{K}$ is satisfiable. A lot of other reasoning tasks (e.g., checking for concept satisfiability wrt. a TBox, entailment of fuzzy ABox assertions, or the BTVB problem) can be reduced to KB satisfiability checking (cnf. [13]) and therefore solved by a respective decision procedure. For this reason, we consider KB satisfiability as the reasoning problem to be solved.

3 Fixpoint-based Decision Procedure

We present a decision procedure for KB satisfiability in $\mathbb{ALC}$ which does not rely on systematic search in the first place (as e.g. tableau-based methods), but instead constructs a canonical interpretation by means of a fixpoint construction. The so-constructed (canonical) interpretation (if non-empty) satisfies the TBox of a KB and allows to derive a model for the given knowledge base $\mathcal{K}$ iff. $\mathcal{K}$ is satisfiable. In contrast to tableau-based procedures a canonical interpretation is in general *not* tree-shaped. Further, it can be shown that the number of iterations required to reach a fixpoint is *linear* in the modal depth of $\mathcal{K}$.

Preprocessing. Without loss of generality, we can restrict ourselves to *normalized* knowledge bases, i.e. knowledge bases which contain only fuzzy ABox assertions of the form $\langle \alpha \geq d \rangle$ [12]. Further, we can assume that all axioms in $\mathcal{K}$ are in box normal form (BNF) [6] (i.e. the only negative concept subexpressions are of the form $\neg \forall R.C$ or negated atomic concept names $\neg C$).

3.1 Basic Notions and Intuition

Types. Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ denote a normalized $\mathbb{ALC}$ knowledge base in BNF. Let $\mathsf{sub}(\mathcal{K})$ denote the set of all concept expressions that occur as subexpressions somewhere in an axiom in $\mathcal{K}$. The *closure* of a knowledge base $\mathsf{cl}(\mathcal{K})$ is defined as the smallest set of concept expressions such that for all $C \in \mathsf{sub}(\mathcal{K})$, if C is not of the form $\neg D$, then $\{C, \neg C\} \subseteq \mathsf{cl}(\mathcal{K})$. Further, let $\mathsf{PossDeg}(\mathcal{K})$ denote the set of all relevant possibility degrees that can be derived from $\mathcal{K}$, i.e. $\mathsf{PossDeg}(\mathcal{K}) = \{0, 0.5, 1\} \cup \{d | \langle \alpha \geq d \rangle \in \mathcal{A}\} \cup \{1 - d | \langle \alpha \geq d \rangle \in \mathcal{A}\}$. It has been shown in [13,14] that if $\mathcal{K}$ is satisfiable, then there is as well a model of $\mathcal{K}$ which assigns possibility degrees in $\mathsf{PossDeg}(\mathcal{K})$ only. Hence, for our purposes we do not need to consider arbitrary possibility degrees $d \in [0, 1]$, but only the *finite* set $\mathsf{PossDeg}(\mathcal{K})$ that can be derived from $\mathcal{K}$.

The closure $\mathsf{cl}(\mathcal{K})$ and the relevant possibility degrees $\mathsf{PossDeg}(\mathcal{K})$ together give us the basic vocabulary to describe individuals and their (fuzzy) properties in interpretations for $\mathcal{K}$. More specifically, the notion of a *type* allows to represent individuals of an interpretation in a syntactic way: a *fuzzy $\mathcal{K}$-type* τ is a maximal subset of $\mathsf{cl}(\mathcal{K}) \times \mathsf{PossDeg}(\mathcal{K})$ such that:

1. if $\langle C, d \rangle \in \tau$ and $\langle C, d' \rangle \in \tau$ then $d = d'$
2. if $C = \neg C'$ then $\langle C, d \rangle \in \tau$ iff. $\langle C', 1 - d \rangle \in \tau$
3. if $C = C' \sqcap C''$ then $\langle C, d \rangle \in \tau$ iff. $\langle C', d' \rangle \in \tau$ and $\langle C'', d'' \rangle \in \tau$ and $d = min(d', d'')$
4. if $C = C' \sqcup C''$ then $\langle C, d \rangle \in \tau$ iff. $\langle C', d' \rangle \in \tau$ and $\langle C'', d'' \rangle \in \tau$ and $d = max(d', d'')$
5. for all $C \sqsubseteq C' \in \mathcal{T}$: if $\langle C, d \rangle \in \tau$ and $\langle C', d' \rangle \in \tau$ then $d \leq d'$
6. if $C = \top$ then $\langle C, 1 \rangle \in \tau$

Since $\mathsf{cl}(\mathcal{K})$ and $\mathsf{PossDeg}(\mathcal{K})$ are both finite sets, there are at most $2^{|\mathsf{cl}(D)| \cdot |\mathsf{PossDeg}(\mathcal{K})|}$

different $\mathcal{K}$-types. Each type τ can be seen as an individual and syntactically represents *all* (fuzzy) properties that can be observed about that individual: $\langle C, d\rangle \in \tau$ represents the statement that the respective individual τ belongs to concept C with the possibility degree d. Hence, the set of all $\mathcal{K}$-types (or simply types) provides enough vocabulary to let us describe all kinds of interpretations for $\mathcal{K}$ simply be fixing how to interconnect individuals (and therefore types).

Canonical Model. It turns out that it is possible to connect types in a fixed (or canonical) way, such that the interconnection defined is consistent with *almost* all properties specified syntactically in the type. The interconnections can be derived from the types themselves:

For a set of types T we can define for each role R a *canonical accessibility relation* $\Delta_R : T \times T \rightarrow \mathsf{PossDeg}(\mathcal{K})$ that "maximally" interconnects types $\tau, \tau' \in T$ with possibility degree $d \in \mathsf{PossDeg}(\mathcal{K})$: Let $\delta(d, d') := 1$ if $d \leq d'$ and $\delta(d, d') := 1 - d$ if $d > d'$. Then, we can define Δ_R by

$$\Delta_R(\tau, \tau') := min\{\delta(d, d') | \langle \forall R.C, d\rangle \in \tau, \langle C, d'\rangle \in \tau'\}$$

if $\forall R.C \in \mathsf{cl}(\mathcal{K})$ for some $C \in \mathbf{C}$, and $\Delta_R(\tau, \tau') := 1$ otherwise.

This way, we can construct a canonical interpretation $\mathcal{I}_T$ for any given set of types T using the canonical interconnection of types by Δ_R as follows: $\mathcal{I}_T = (T, \cdot^{\mathcal{I}_T})$ with (i) for any (atomic) concept name C in $\mathcal{K}$ and any $\tau \in T$ we set $C^{\mathcal{I}_T}(\tau) = d$ if $\langle C, d\rangle \in \tau$, and (ii) $R^{\mathcal{I}_T}(\tau, \tau') = \Delta_R(\tau, \tau')$ for any role R in $\mathcal{K}$ and any $\tau, \tau' \in T$. Please note, that by our definition of $\mathcal{K}$-types, $\mathcal{I}_T$ is well-defined for any concept name or role name. However, our definition deliberately leaves open the interpretation of individuals. We therefore define in fact a class of canonical interpretations, each of which fixes a specific way of how to interpret the individuals in a KB $\mathcal{K}$.

The canonical interconnection in $\mathcal{I}_T$ is chosen in such a way that all assignments of possibility degrees to concepts of the form $C = \forall R.C \in \tau$ are lower bounds for the possibility degrees that are in fact assigned by a canonical interpretation $\mathcal{I}_T$. Hence, such a canonical interpretation is *almost* immediately a (canonical) model for the terminology $\mathcal{T}$, i.e. it satisfies that

$$C^{\mathcal{I}_T}(\tau) = d \text{ iff. } \langle C, d\rangle \in \tau \qquad (*)$$

for *almost* all $C \in \mathsf{cl}(\mathcal{K})$ it holds and therefore $\mathcal{I}_T \models C \sqsubseteq C'$ for all $C \sqsubseteq C' \in \mathcal{T}$ by clause (5) in our definition of $\mathcal{K}$-types. That $(*)$ is satisfied by $\mathcal{I}_T$ is straightforward for the cases of concept names C, or complex concepts of the form $C = C' \sqcap C''$, $C = C' \sqcup C''$, $C = \neg C'$ and the $C^{\mathcal{I}_T}(\tau) \geq d$ case for $C = \forall R.C$ by our definition of types and the definition of Δ_R. The only cases where $(*)$ can be violated by $\mathcal{I}_T$ is for types τ containing universally role restricted concepts $\forall R.C$ that are assigned a possibility degree which is *too small* (wrt. the R-successor types τ' in $\mathcal{I}_T$) to properly reflect the semantics of $\forall R.C$ in $\mathbb{ALC}$, i.e. to coincide with the *greatest* lower bound of the set

$$\{max(1 - R^{\mathcal{I}_T}(\tau, \tau'), C^{\mathcal{I}_T}(\tau')) \,|\, \tau' \in T\}$$

Types τ in which the possibility degree assigned d to $\forall R.C$ is too small to be consistent with the semantics of $\mathbb{ALC}$ are called *bad types*. Bad types $\tau \in T$ can be detected

easily, since they satisfy that there exist $R \in \mathbf{R}, C \in \mathcal{C}(\Sigma), d \in \mathsf{PossDeg}(\mathcal{K})$ s.t. $\langle \forall R.C, d\rangle \in \tau$ and for all $\tau' \in T$: if $\langle C, d'\rangle \in \tau'$ then $max(1 - \Delta_R(\tau, \tau'), d') > d$.

This suggest the the following simple algorithm (which uses a *fuzzy type elimination* process as its core): in order to compute a maximal interpretation that satisfies all terminological axioms, we start off with the maximal set of types (i.e all $\mathcal{K}$-types) and iteratively fix all problems that prevent $(*)$ from being satisfied by removing bad types. This way, we must eventually reach a fixpoint after finitely many steps. If the resulting set of types is non-empty, we know that $(*)$ must hold (since all problems have been fixed) and therefore we can be certain that the corresponding canonical interpretation satisfies $\mathcal{T}$ (and covers all other possible models of $\mathcal{T}$ at the same time). Hence, we eventually need to check if all ABox axioms are satisfied by the canonical interpretation. If this is the case, we have found a model for $\mathcal{K}$, otherwise, we know that there can not be any interpretation that satisfies both $\mathcal{T}$ and $\mathcal{A}$ at the same time. In other words, $\mathcal{K}$ is not satisfiable.

Algorithm. The type elimination process sketched above can be formalized as shown in Fig. 1. Note that the emptiness test for the fixpoint T is covered implicitly: if the fixpoint T is empty, then the test in the if-statement fails trivially.

procedure satisfiable($\mathcal{K}$): boolean
T := $\{\tau | \tau$ is a $\mathcal{K}$-type $\}$;
repeat
  $T' := T$;
  $T := T' \setminus badtypes(T')$;
until $T = T'$;
if *there exists a total function* $\pi : Ind_{\mathcal{A}} \rightarrow T$ *s.t.* $\langle C, d'\rangle \in \pi(o)$ *and* $d \leq d'$ *for each* $\langle o : C \geq d\rangle \in \mathcal{A}$*, and* $\Delta_R(\pi(o), \pi(o')) \geq d$ *for each* $\langle R(o, o') \geq d\rangle \in \mathcal{A}$ **then**
  return true;
end
return false;

function $badtypes(T) : 2^T$
return $\{\tau \in T | \langle \forall R.C, d\rangle \in \tau$ and for all $\tau' \in T$: if $\langle C, d'\rangle \in \tau'$ then $max(1 - \Delta_R(\tau, \tau'), d') > d\}$;

Algorithm 1: The Type Elimination-based Decision Procedure $\mathbf{FixIt}(\mathbb{ALC})$

3.2 Soundness, Completeness and Termination

The termination, soundness, and completeness of our algorithm can be proven formally.

Theorem 1 (Termination). *For any* $\mathbb{ALC}$ *knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ *the algorithm* $\mathbf{FixIt}(\mathbb{ALC})$ *terminates after finitely many steps with either* true *or* false *as return value.*

Proof. The initialization step of the algorithm takes finitely many steps since the number of $\mathcal{K}$-types is finite. The repeat-loop must terminate after finitely many steps, since we start with a finite set of types T in the beginning: if we do not remove any type in an iteration (i.e. $badtypes(T') = \emptyset$) we have $T = T'$ at the end of the loop (i.e. reaching a fixpoint) and therefore terminate the loop. On the other hand, if $badtypes(T') \neq \emptyset$ in an iteration, at least one type is removed from T' and hence $T \subset T'$. This means, that the input set of types T for the next iteration is finite and strictly smaller. Clearly, the empty set is a fixpoint of $badtypes(\cdot)$ too, i.e. $badtypes(\emptyset) = \emptyset$. Hence, we can repeat the loop only finitely many times until we finally will reach a fixpoint. Since this fixpoint T is a subset of the finite set of the initial set of types and there are only finitely many possible mappings π to consider, deciding the criterion in the if-statement (based on T) takes as well only finitely many steps. Therefore, the algorithm terminates with one of the return-statements that give as a result either true *or* false.

The following lemma is a key element of the soundness and completeness proof and shows that by successively removing bad types we can indeed ensure that types encode possibility degree assignments to concepts that coincide with the canonical interpretation, and that any such canonical interpretation is a model of the $\mathcal{T}$.

Let T be the set of types that is computed as the fixpoint in the algorithm $\mathbf{FixIt}(\mathbb{ALC})$, i.e. $badtypes(T) = \emptyset$ and let $\mathcal{I}_T$= $(T, \cdot^{\mathcal{I}_T})$ be a canonical interpretation for T as defined above.

Lemma 1. *For each $\mathcal{K}$-type τ, concept $C \in \mathsf{cl}(\mathcal{K})$ and $d \in \mathsf{PossDeg}(\mathcal{K})$ it holds that $C^{\mathcal{I}_T}(\tau) = d$ iff. $\langle C, d\rangle \in \tau$. Further, $\mathcal{I}_T \models \mathcal{T}$.*

Proof. For the first part of the lemma, let τ be any $\mathcal{K}$-type and $d \in \mathsf{PossDeg}(\mathcal{K})$ be any relevant possibility degree.

We show by induction over the structure of concepts $C \in \mathsf{cl}(\mathcal{K})$ that $\langle C, d\rangle \in \tau$ iff. $C^{\mathcal{I}_T}(\tau) = d$: the base case (i.e. $C \in \mathsf{cl}(\mathcal{K})$ is an atomic concept name $C \in \mathbf{C}$ or $C = \top$) is trivially satisfied by our definition of $\mathcal{I}_T$. For the induction step, we consider the different cases of compound concept expressions $C \in \mathsf{cl}(\mathcal{K})$ one-by-one:

1. for $C = C_1 \sqcap C_2 \in \mathsf{cl}(\mathcal{K})$, we know that $C_1, C_2 \in \mathsf{cl}(\mathcal{K})$. By clause (3) in our definition of types, we know that $\langle C, d\rangle = \langle C_1 \sqcap C_2, d\rangle \in \tau$ iff. $\langle C_1, d_1\rangle \in \tau$ and $\langle C_2, d_2\rangle \in \tau$ and $d = min(d_1, d_2)$. Applying the induction hypothesis to C_1 and C_2, we know that this is the case iff. ${C_1}^{\mathcal{I}_T}(\tau) = d_1$ and ${C_2}^{\mathcal{I}_T}(\tau) = d_2$ and $d = min(d_1, d_2)$ iff. $d = min({C_1}^{\mathcal{I}_T}(\tau), {C_2}^{\mathcal{I}_T}(\tau))$ iff. $d = (C_1 \sqcap C_2)^{\mathcal{I}_T} = C^{\mathcal{I}_T}$ by the by the semantics of $\sqcap$ in $\mathbb{ALC}$.

2. for $C = C_1 \sqcup C_2 \in \mathsf{cl}(\mathcal{K})$ the proof is analogous.

3. for $C = \neg D \in \mathsf{cl}(\mathcal{K})$, we know that $D \in \mathsf{cl}(\mathcal{K})$ by the definition of $\mathsf{cl}(\mathcal{K})$. Because of clause (2) and the maximality requirement in our definition of $\mathcal{K}$-types, we know that $\langle C, d\rangle = \langle \neg D, d\rangle \in \tau$ iff. $\langle D, 1 - d\rangle \in \tau$. Applying the induction hypothesis for D, we know that this holds iff. $D^{\mathcal{I}_T}(\tau) = 1 - d$ iff. $(\neg D)^{\mathcal{I}_T}(\tau) = C^{\mathcal{I}_T}(\tau) = d$ by the semantics of concept negation in $\mathbb{ALC}$.

4. for $C = \forall R.D \in \mathsf{cl}(\mathcal{K})$, $D \in sub(\mathcal{K})$ holds and hence $D \in \mathsf{cl}(\mathcal{K})$ by the definition of $\mathsf{cl}(\mathcal{K})$.

First, we show one direction, i.e. that $C^{\mathcal{I}_T}(\tau) = d$ if $\langle C, d\rangle \in \tau$: Assume that $\langle C, d\rangle = \langle \forall R.D, d\rangle \in \tau$. According to the semantics of the universal role restriction in $\mathbb{ALC}$ and our definition of $\mathcal{I}_T$, we have $C^{\mathcal{I}_T}(\tau) = (\forall R.D)^{\mathcal{I}_T}(\tau) = inf_{\tau' \in T}\{max(1 - R^{\mathcal{I}_T}(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\} = inf_{\tau' \in T}\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\}$. We show that d is a lower bound for $\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau')) | \tau' \in T\}$: Assume there exists a $\tau' \in T$ s.t. $d > max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))$. Let $D^{\mathcal{I}_T}(\tau') = d'$. Applying the induction hypothesis to $D \in \mathsf{cl}(\mathcal{K})$, we know $\langle D, d'\rangle \in \tau'$. Hence, both $d > 1 - \Delta_R(\tau, \tau')$ and $d > d'$ must hold. Hence $\Delta_R(\tau, \tau') > 1 - d$. But, since $\langle \forall R.D, d\rangle \in \tau$ this is not possible by our by our definition of Δ_R, because $\Delta_R(\tau, \tau') \leq 1 - d$. From the contradiction we can conclude that d is in fact a lower bound for the considered set. Therefore, $d \leq inf_{\tau' \in T}\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\}$

Next, we show that $inf_{\tau' \in T}\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\} \leq d$ too, by proving that there exists a $\tau' \in T$ s.t. $max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau')) \leq d$. Assume, the contrary, i.e. for all $\tau' \in T$: $max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau')) > d$ (‡). By applying our induction hypothesis to D and τ', we know that this is the case iff. for all $\tau' \in T$: if $\langle D, d'\rangle \in \tau'$ then $max(1 - \Delta_R(\tau, \tau'), d') > d$. But then, τ would be a bad type which contradicts the fact that T the computed fixpoint which can not contain any bad types (i.e. $\tau \in badtypes(T) = \emptyset$). Hence our assumption (‡) must be wrong, and we can conclude that $inf_{\tau' \in T}\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\} \leq d$. Therefore, $d = inf_{\tau' \in T}\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\}$, and hence $d = C^{\mathcal{I}_T}(\tau)$.

The other direction of the induction hypothesis (i.e. that $\langle \forall R.D, d\rangle \in \tau$ if $(\forall R.D)^{\mathcal{I}_T}(\tau) = d$) can now be proven as follows: Assume that $(\forall R.D)^{\mathcal{I}_T}(\tau) = d$ (†) but $\langle \forall R.D, d\rangle \notin \tau$. By the maximality requirement in our definition of $\mathcal{K}$-types there must hence exist a $d' \in \mathsf{PossDeg}(\mathcal{K})$ s.t. $\langle \forall R.D, d'\rangle \in \tau$ and $d' \neq d$. Using the same argument as for the if-direction in this case, we can therefore conclude that $(\forall R.D)^{\mathcal{I}_T}(\tau) = d' \neq d$ which contradicts (†). Hence, our assumption must be wrong and $\langle \forall R.D, d\rangle \in \tau$ must hold whenever $(\forall R.D)^{\mathcal{I}_T}(\tau) = d$.

For the second part of the lemma, to show that $\mathcal{I}_T \models \mathcal{T}$, assume that for some $\alpha = C \sqsubseteq C' \in \mathcal{T}$ and some $\tau \in T$ it holds that $C^{\mathcal{I}_T}(\tau) > C'^{\mathcal{I}_T}(\tau)$, in other words, if $C^{\mathcal{I}_T}(\tau) = d$ and $C'^{\mathcal{I}_T}(\tau) = d'$ then $d > d'$. Thus, we can deduce (from the first part of this lemma) that, if $\langle C, d\rangle \in \tau$ and $\langle C', d'\rangle \in \tau$ then $d > d'$. However, by our definition of $\mathcal{K}$-type (i.e. clause (5)), we also know that in this case $d \leq d'$ must hold, which is contradictive. Hence, our assumption must be wrong and $\mathcal{I}_T \models \alpha$ for each $\alpha \in \mathcal{T}$ which means that $\mathcal{I}_T \models \mathcal{T}$.

Theorem 2 (Soundness). *If* $\mathbf{FixIt}(\mathbb{ALC})$ *returns true for a $\mathbb{ALC}$ knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, then $\mathcal{K}$ is satisfiable.*

Proof. We show that a canonical interpretation $\mathcal{I}_T$ for the computed fixpoint T can be extended to a model of $\mathcal{K}$. By Lemma 1, we already know that $\mathcal{I}_T \models \mathcal{T}$. We now show, that $\mathcal{I}_T$ can be extended such that $\mathcal{I}_T \models \mathcal{A}$ too, which completes the proof: Since the algorithm returns true, there exist a total function $\pi : Ind_{\mathcal{A}} \to T$ s.t. $\langle C, d'\rangle \in \pi(o)$ and $d \leq d'$ for each $\langle o : C \geq d\rangle \in \mathcal{A}$ ($\star$), and $\Delta_R(\pi(o), \pi(o')) \geq d$ for each $\langle R(o, o') \geq d\rangle \in \mathcal{A}$ (†). We extend the definition of $\mathcal{I}_T$ to the ABox $\mathcal{A}$ as follows: for all Abox individual names $o \in Ind_{\mathcal{A}}$, we set $o^{\mathcal{I}_T} := \pi(o) \in T$. First, consider an ABox axiom of the form $\alpha = \langle o : C \geq d\rangle \in \mathcal{A}$. Then, $\mathcal{I}_T \models \alpha$ iff. $C^{\mathcal{I}_T}(o^{\mathcal{I}_T}) \geq d$,

iff. $C^{\mathcal{I}_T}(\pi(o)) \geq d$ *iff. there exists a* $d' \geq d$ *s.t.* $C^{\mathcal{I}_T}(\pi(o)) = d'$. *By Lemma 1 this is the case iff. there exists a* $d' \geq d$ *s.t.* $\langle C, d' \rangle \in \pi(o)$ *which is satisfied since* ($\star$) *holds. Second, consider an ABox axiom of the form* $\alpha = \langle R(o, o') \geq d \rangle \in \mathcal{A}$. *Then,* $\mathcal{I}_T \models \alpha$ *iff.* $R^{\mathcal{I}_T}(o^{\mathcal{I}_T}, o'^{\mathcal{I}_T}) \geq d$ *iff.* $\Delta_R(\pi(o), \pi(o')) \geq d$ *(by Def. of the extended* $\mathcal{I}_T$*). The latter is satisfied because of* (†).

An second key element for the completeness proof is the following lemma that shows that our canonical way of interconnecting types (in the fixpoint set) is maximal or the strongest possible one in the following sense: the interconnection R of individuals o, o' defined by any model $\mathcal{I}$ of $\mathcal{K}$ is covered by the canonical interconnection Δ_R of the respective types $\tau(o), \tau(o')$ representing o, o' in $\mathcal{I}$.

Lemma 2. *Let* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *be any model of* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$. *For each individual* $o \in \Delta^{\mathcal{I}}$ *we define its corresponding type* $\tau(o) := \{\langle C, d \rangle \in \mathsf{cl}(\mathcal{K}) \times \mathsf{PossDeg}(\mathcal{K}) | C^{\mathcal{I}}(o) = d\}$. *Then,* $\Delta_R(\tau(o), \tau(o')) \geq R^{\mathcal{I}}(o, o')$ *for all* $o, o' \in \Delta^{\mathcal{I}}$.

Proof. Assume that there exist $o, o' \in \Delta^{\mathcal{I}}$ *s.t.* $\Delta_R(\tau(o), \tau(o')) < R^{\mathcal{I}}(o, o')$. *By our definition of* Δ_R, *we then know that* $\delta(d, d') < R^{\mathcal{I}}(o, o')$ $(*)$ *for some* $\langle \forall R.C, d \rangle \in \tau(o)$ *and* $\langle C, d' \rangle \in \tau(o')$. *From the definition of* $\tau(\cdot)$, *we know that* $\delta(d, d') < R^{\mathcal{I}}(o, o')$ *for some* $o, o' \in \Delta^{\mathcal{I}}$ *s.t.* $(\forall R.C)^{\mathcal{I}}(o) = d$ *and* $C^{\mathcal{I}}(o') = d'$. *From the semantics of* $\forall R.C$ *in* $\mathbb{ALC}$, *we derive* $d = inf_{o'' \in \Delta^{\mathcal{I}}}\{max(1 - R^{\mathcal{I}}(o, o''), C^{\mathcal{I}_T}(o''))\}$. *Hence, in particular* $d \leq max(1 - R^{\mathcal{I}}(o, o'), C^{\mathcal{I}_T}(o')) = max(1 - R^{\mathcal{I}}(o, o'), d')$ (†). *We consider two cases: first,* $d' < d$, *then in order to satisfy* (†), $max(1 - R^{\mathcal{I}}(o, o'), d') = 1 - R^{\mathcal{I}}(o, o')$ *must hold and* (†) *simplifies to* $d \leq 1 - R^{\mathcal{I}}(o, o')$ *iff.* $R^{\mathcal{I}}(o, o') \leq 1 - d$. *Since* $d' < d$, $(*)$ *simplifies to* $1 - d < R^{\mathcal{I}}(o, o')$, *hence,* $1 - d < R^{\mathcal{I}}(o, o') \leq 1 - d$ *which is contradictory. In the second case, we assume that* $d' \geq d$. *Hence,* $\delta(d, d') = 1$. *Then,* (†) *simplifies to* $1 < R^{\mathcal{I}}(o, o')$, *which is contradictory, since* $R^{\mathcal{I}}(o, o') \in \mathsf{PossDeg}(\mathcal{K})$ *and* 1 *is the maximum possibility degree in* $\mathsf{PossDeg}(\mathcal{K})$. *Therefore, in both cases we reach a contradiction and can conclude that our assumption must be wrong. This concludes the proof.*

Theorem 3 (Completeness). *If an* $\mathbb{ALC}$ *knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ *is satisfiable, then* $\mathbf{FixIt}(\mathbb{ALC})$ *returns true for* $\mathcal{K}$

Proof. Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *be any model of* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, *i.e.* $\mathcal{I} \models \mathcal{T}$ *and* $\mathcal{I} \models \mathcal{A}$. *In [13,14] it is shown that a KB in* $\mathbb{ALC}$ *is consistent iff. there is a model* $\mathcal{I}$ *of the KB which only assigns possibility degrees that occur in the ABox* $\mathcal{A}$ *for interpreting atomic concept or role names. Hence, without loss of generality, we can assume in the following that* $\mathcal{I}$ *assigns possibility degrees in* $\mathsf{PossDeg}(\mathcal{K})$ *only.*

For each individual $o \in \Delta^{\mathcal{I}}$ *we define its corresponding type* $\tau(o) := \{\langle C, d \rangle \in \mathsf{cl}(\mathcal{K}) \times \mathsf{PossDeg}(\mathcal{K}) | C^{\mathcal{I}}(o) = d\}$ *and define* $T_{\mathcal{I}} := \{\tau(o) | o \in \Delta^{\mathcal{I}}\}$. *It is easy to see that* $T_{\mathcal{I}}$ *is a set of* $\mathcal{K}$*-types. Further,* $T_{\mathcal{I}} \neq \emptyset$ *since* $\Delta^{\mathcal{I}} \neq \emptyset$.

Let $T^{(i)}$ *denote the set of types that is computed after* i *iterations of the repeat-loop in our algorithm. We first show that* $T_{\mathcal{I}} \subseteq T^{(i)}$ *for all* $i \geq 0$ *by induction over the number of iterations* i*:*

In the base case $i = 0$, *our initialization step sets* $T^{(0)}$ *to contain all* $\mathcal{K}$*-types. Since* $T_{\mathcal{I}}$ *consists of* $\mathcal{K}$*-types only,* $T_{\mathcal{I}} \subseteq T^{(0)}$ *must hold. To proof the induction step, we*

assume that $T_\mathcal{I} \subseteq T^{(i)}$ but $T_\mathcal{I} \not\subseteq T^{(i+1)}$. Therefore, there must be a $\tau(o) \in T_\mathcal{I}$ s.t. $\tau(o) \in T^{(i)}$ but not $\tau(o) \in T^{(i+1)}$. From the repeat-loop in the algorithm, we know that $T^{(i+1)} = T^{(i)}\ badtypes(T^{(i)})$. Consequently, $\tau(o)$ must be a bad type $\tau(o) \in badtypes(T^{(i)})$ and we can not have reached a fix-point yet.

From our definition of bad-types we can derive that there must exist a $\langle \forall R.C, d\rangle \in \tau(o)$ and for all $\tau' \in T^{(i)}$: if $\langle C, d'\rangle \in \tau'$ then $max(1 - \Delta_R(\tau, \tau'), d') > d\}$ (‡). Since $T_\mathcal{I} \subseteq T^{(i)}$ (‡) must hold in particular for all $\tau(o') \in T_\mathcal{I}$. Using our definition of $\tau(\cdot)$ we can rephrase (‡) as follows: there must exist a $\forall R.C \in \mathsf{cl}(\mathcal{K})$ s.t. for all $o' \in \Delta^\mathcal{I}$: $max(1 - \Delta_R(\tau(o), \tau(o')), C^\mathcal{I}(o')) > (\forall R.C)^\mathcal{I}(o)$ ($\star$).

By Lemma 2, we know that $\Delta_R(\tau(o), \tau(o')) \geq R^\mathcal{I}(o, o')$ for all $o, o' \in \Delta^\mathcal{I}$. Hence, $1 - \Delta_R(\tau(o), \tau(o')) \leq 1 - R^\mathcal{I}(o, o')$ ($*$) for all $o, o' \in \Delta^\mathcal{I}$. Since $max(a, b) \leq max(a', b)$ for any a, a', b s.t. $a \leq a'$, we can reformulate ($\star$) using ($*$) as follows: there must exist a $\forall R.C \in \mathsf{cl}(\mathcal{K})$ s.t. for all $o' \in \Delta^\mathcal{I}$: $max(1 - R^\mathcal{I}(o, o'), C^\mathcal{I}(o')) > (\forall R.C)^\mathcal{I}(o)\}$ ($\natural$), which contradicts the fact that $(\forall R.C)^\mathcal{I}(o) = inf_{o' \in \Delta^\mathcal{I}}\{max(1 - R^\mathcal{I}(o, o'), C^\mathcal{I}(o'))\}$: Since $R^\mathcal{I}(o, o') \in \mathsf{PossDeg}(\mathcal{K})$ and $C^\mathcal{I}(o') \in \mathsf{PossDeg}(\mathcal{K})$, we know that $max(1 - R^\mathcal{I}(o, o'), C^\mathcal{I}(o')) \in \mathsf{PossDeg}(\mathcal{K})$ for all $o' \in \Delta^\mathcal{I}$ by our definition of $\mathsf{PossDeg}(\mathcal{K})$. Because $\Delta^\mathcal{I} \neq \emptyset$ and $\mathsf{PossDeg}(\mathcal{K})$ is a finite set, there must exist an $o^* \in \Delta^\mathcal{I}$ for which $max(1 - R^\mathcal{I}(o, o^*), C^\mathcal{I}(o^*))$ is minimal, i.e. $max(1 - R^\mathcal{I}(o, o^*), C^\mathcal{I}(o^*)) \leq max(1 - R^\mathcal{I}(o, o'), C^\mathcal{I}(o'))$ for all $o' \in \Delta^\mathcal{I}$. Hence, $d^* := max(1 - R^\mathcal{I}(o, o^*), C^\mathcal{I}(o^*)) \in \mathsf{PossDeg}(\mathcal{K})$ is a lower bound for the set $\{max(1 - R^\mathcal{I}(o, o'), C^\mathcal{I}(o') | o' \in \Delta^\mathcal{I}\}$. However, from ($\natural$) we know that $d < d^*$, hence d can not be the greatest lower bound (i.e. the infimum) of the set $\{max(1 - R^\mathcal{I}(o, o'), C^\mathcal{I}(o') | o' \in \Delta^\mathcal{I}\}$, hence $\forall R.C)^\mathcal{I}(o) = d^* > d$ which is contradictive.

Therefore, our assumption that $\tau(o)$ is a bad type must be wrong and we have completed the proof of the induction step as well as the induction argument.

We continue the proof of the lemma as follows: since $T = T^{(i)}$ is the fixpoint that is computed in the loop in our algorithm in i steps for some $i \geq 0$, we know that $T_\mathcal{I} \subseteq T^{(i)} = T$. Consider the mapping $\pi_\mathcal{I} : Ind_\mathcal{A} \to T$ defined by $\pi_\mathcal{I}(o) := \tau(o^\mathcal{I})$ for all $o \in Ind_\mathcal{A}$. Then, $\pi_\mathcal{I}$ is a well-defined, total function from $Ind_\mathcal{A}$ to T. We now show that this specific mapping $\pi_\mathcal{I}$ satisfies the condition that is checked in the if-statement of the algorithm:

In the first case, we consider any Abox axiom $\alpha \in \mathcal{A}$ of the form $\alpha = \langle o : C \geq d\rangle$. Since $\mathcal{I} \models \mathcal{A}$, $\mathcal{I} \models \alpha$ must hold. $\mathcal{I} \models \alpha$ iff. $C^\mathcal{I}(o^\mathcal{I}) \geq d$ iff. $C^\mathcal{I}(o^\mathcal{I}) = d'$ for some $d' \in \mathsf{PossDeg}(\mathcal{K})$ with $d' \geq d$ iff. $\langle C, d\rangle \in \tau(o^\mathcal{I})$ for some $d' \in \mathsf{PossDeg}(\mathcal{K})$ with $d' \geq d$ (by Lemma 1) iff. $\langle C, d\rangle \in \pi_\mathcal{I}(o)$ for some $d' \in \mathsf{PossDeg}(\mathcal{K})$ with $d' \geq d$ (by our definition of $\pi_\mathcal{I}$). Hence, the respective part of the if-condition for α holds for $\pi_\mathcal{I}$. In the second case, we consider any Abox axiom $\alpha \in \mathcal{A}$ of the form $\alpha = \langle R(o, o') \geq d\rangle$. Since, $\mathcal{I} \models \mathcal{A}$, $\mathcal{I} \models \alpha$ must hold. $\mathcal{I} \models \alpha$ holds iff. $R^\mathcal{I}(o^\mathcal{I}, o'^\mathcal{I}) \geq d$. Since $\Delta_R(\tau(o), \tau(o')) \geq R^\mathcal{I}(o^\mathcal{I}, o'^\mathcal{I})$ by Lemma 2, we know that $\Delta_R(\tau(o^\mathcal{I}), \tau(o'^\mathcal{I})) \geq d$ and therefore $\Delta_R(\pi_\mathcal{I}(o), \pi_\mathcal{I}(o)) \geq d$ by our definition of $\pi_\mathcal{I}$. Hence, the respective part of the if-condition for α is as well satisfied by $\pi_\mathcal{I}$. Consequently, the tested if-condition is satisfied by $\pi_\mathcal{I}$ and the algorithm returns true.

This leads to the main result, which is an immediate consequence of Theorems 2, 3, and 1:

Corollary 1. *The algorithm* $\mathbf{FixIt}(\mathbb{ALC})$ *is a sound and complete decision procedure for knowledge base satisfiability in* $\mathbb{ALC}$.

4 Related Work

Our method $\mathbf{FixIt}(\mathbb{ALC})$ generalizes the principle (i.e. a type elimination process) underlying the top-down variant of the $\mathcal{KBDD}$ procedure proposed in [6] for the modal logic $\mathbf{K}$ to the (more expressive) FDL $\mathbb{ALC}$. Further, our method integrates (fuzzy) ABoxes and TBoxes in the inference process both of which are *not* dealt with in $\mathcal{KBDD}$.

So far, reasoning in Fuzzy DLs has been mostly based on tableau-methods (e.g., [13,12,4,11]). Most of these methods do not support reasoning with general terminologies as it is possible with $\mathbf{FixIt}(\mathbb{ALC})$. The first method ever to integrate GCIs into FDL reasoning is [12]. A very similar approach is presented in [4] for the fuzzy variant of a more expressive DL, namely $\mathcal{SHI}$. Very recently, [16] proposed a novel and elegant method for reasoning with GCIs (under a more general semantics than here) which is inspired by earlier works on tableau-based reasoning in multi-valued logics. To the best of our knowledge there is no other approach to deal with GCIs in FDLs available at present. $\mathbf{FixIt}(\mathbb{ALC})$ therefore represents an interesting enrichment of inference calculi toolbox for FDLs, since no non-determinism is introduced by considering GCIs. A similar effect is achieved in [16] by the substantial modification of a standard tableau-based method and an extension with an MILP oracle. A very similar approach to [16] that is not fixed to a specific semantics is presented in [3].

Further, [14] demonstrates how to use inference procedures for *classical* DLs to perform reasoning in (some) FDLs. This allows to use algorithms that have been developed for classical DLs in FDL reasoning (for some FDLs) in an indirect way. Please note that the $\mathcal{KBDD}$ procedure can not be used in such an indirect way to perform $\mathbb{ALC}$ reasoning, since both TBoxes and ABoxes are not supported.

5 Conclusions and Future Work

We presented a novel procedure $\mathbf{FixIt}(\mathbb{ALC})$ for deciding knowledge base (KB) satisfiability in the FDL $\mathbb{ALC}$, introducing a *new class* of inference procedures into FDL reasoning. Besides the tableau-based methods [12,4,16], it is the only (and the first non-tableau-based) approach to integrate general terminologies in FDL reasoning that we are aware of.

The main research questions that we want to address next are as follows: we will study means of implicit representation of sets of fuzzy types known from Symbolic Model Checking, in particular their implementation by means of Ordered Binary Decision Diagrams (OBDDs) similar to [6], therefore addressing the main obstacle to apply the procedure in practice. A major question concerning optimization is clearly how to implement the final test of the algorithm efficiently, e.g. by heuristic search using the information in the ABox effectively to find the required mapping. The integration of optimizations such as full vs. lean representations or particle vs. types as discussed in [6] should be straightforward. We want to evaluate the effectiveness of the method

by an implementation and comparison to tableau-based systems for FDLs. Moreover, we believe that it is interesting to study a bottom-up variant of $\mathcal{KBDD}$ in the context of FDLs too, and to check if the integration of ABoxes can be done more efficiently in such a variant. Finally, we would like to see to what extend the method can cover other semantics for FDLs (e.g. other t-norms) and extended constructs, such as fuzzy modifiers and concrete domains.

References

1. F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.
2. R. E. Bryant. Symbolic Boolean manipulation with ordered binary-decision diagrams. *ACM Comput. Surv.*, 24(3):293–318, 1992.
3. V. Haarslev, H. Pai, and N. Shiri. Uncertainty Reasoning for Ontologies with General TBoxes in Description Logic. In P. C. G. Costa, C. D'Amato, N. Fanizzi, K. B. Laskey, K. Laskey, T. Lukasiewicz, M. Nickles, and M. Pool, editors, *Uncertainty Reasoning for the Semantic Web I*, LNAI. Springer, 2008.
4. Y. Li, B. Xu, J. Lu, and D. Kang. Discrete Tableau Algorithms for $\mathcal{FSHI}$. In *Proceedings of the International Workshop on Description Logics (DL)*, 2006.
5. K. L. McMillan. *Symbolic Model Checking*. Kluwer Academic Publishers, Norwell, MA, USA, 1993.
6. G. Pan, U. Sattler, and M. Y. Vardi. BDD-based decision procedures for the modal logic K. *Journal of Applied Non-Classical Logics*, 16(1-2):169–208, 2006.
7. P. F. Patel-Schneider, P. Hayes, and I. Horrocks. OWL Web Ontology Language Semantics and Abstract Syntax. Candidate Recommendation 18 August 2003, W3C, 2003.
8. V. R. Pratt. A Near-Optimal Method for Reasoning about Action. *J. Comput. Syst. Sci.*, 20(2):231–254, 1980.
9. K. Schild. A correspondence theory for terminological logics: Preliminary report. In *In Proceedings of the International Joint Conference of Artificial Intelligence (IJCAI 1991)*, pages 466–471, 1991.
10. M. Schmidt-Schauß and G. Smolka. Attributive Concept Descriptions with Complements. *Artif. Intell.*, 48(1):1–26, 1991.
11. G. Stoilos, G. B. Stamou, J. Z. Pan, V. Tzouvaras, and I. Horrocks. Reasoning with very expressive fuzzy description logics. *J. Artif. Intell. Res. (JAIR)*, 30:273–320, 2007.
12. G. Stoilos, U. Straccia, G. Stamou, and J. Pan. General Concept Inclusions in Fuzzy Description Logics. In *Proceedings of the 17th Eureopean Conference on Artificial Intelligence (ECAI-06)*, pages 457–461. IOS Press, 2006.
13. U. Straccia. Reasoning within Fuzzy Description Logics. *Journal of Artificial Intelligence Research*, 14:137–166, 2001.
14. U. Straccia. Transforming Fuzzy Description Logics into Classical Description Logics. In *Proceedings of the 9th European Conference on Logics in Artificial Intelligence (JELIA-04)*, number 3229 in Lecture Notes in Computer Science, pages 385–399, Lisbon, Portugal, 2004. Springer Verlag.
15. U. Straccia. A Fuzzy Description Logic for the Semantic Web. In E. Sanchez, editor, *Fuzzy Logic and the Semantic Web*, Capturing Intelligence, chapter 4, pages 73–90. Elsevier, 2006.
16. U. Straccia and F. Bobillo. Mixed integer programming, general concept inclusions and fuzzy description logics. *Mathware & Soft Computing*, 14(3):247–259, 2007.