Reasoning with contextual defeasible $\mathcal{ALC}$

Katarina Britz[1] and Ivan Varzinczak[2,1]

[1] CAIR, Stellenbosch University, South Africa
abritz@sun.ac.za
[2] CRIL, Université d'Artois & CNRS, France
varzinczak@cril.fr

Abstract. In recent work, we addressed an important limitation in previous extensions of description logics to represent defeasible knowledge, namely the restriction in the semantics of defeasible concept inclusion to a single preference order on objects of the domain. Syntactically, this limitation translates to a context-agnostic notion of defeasible subsumption, which is quite restrictive when it comes to modelling different nuances of defeasibility. Our point of departure in our recent proposal allows for different orderings on the interpretation of roles. This yields a notion of contextual defeasible subsumption, where the context is informed by a role. In the present paper, we extend this work to also provide a proof-theoretic counterpart and associated results. We define a (naïve) tableau-based algorithm for checking preferential consistency of contextual defeasible knowledge bases, a central piece in the definition of other forms of contextual defeasible reasoning over ontologies, notably contextual rational closure.

Keywords: description logics · defeasible reasoning · contexts · tableaux.

1 Introduction

Description logics (DLs) [1] are central to many modern AI and database applications since they provide the logical foundation of formal ontologies. Yet, as classical formalisms, DLs do not allow for the proper representation of and reasoning with defeasible information, as shown up in the following example from the access-control domain: employees have access to classified information; interns (who are also employees) do not; but graduate interns do. From a naïve (classical) formalisation of this scenario, one concludes that the class of interns is empty (just as that of graduate interns). But while concept unsatisfiability has been investigated extensively in ontology debugging and repair, our research problem here goes beyond that.

The past 25 years have witnessed many attempts to introduce defeasible-reasoning capabilities in a DL setting, usually drawing on a well-established body of research on non-monotonic reasoning (NMR). These comprise the so-called preferential approaches [13–15, 23, 24, 22, 26, 27, 29, 30, 38, 39], circumscription-based ones [6, 7, 40], as well as others [2, 3, 5, 25, 31–33, 36, 37, 42].

Preferential extensions of DLs turn out to be particularly promising. There a notion of *defeasible subsumption* $\sqsubset\!\!\!\sim$ is introduced, the intuition of a statement of the form $C \sqsubset\!\!\!\sim D$ being that "usually, C is subsumed by D" or "the normal Cs are Ds". The semantics is in terms of an ordering on the set of objects allowing us to identify the most normal elements in C with the *minimal* C-instances w.r.t. the ordering.

The assumption of a single ordering on the domain of interpretation does not allow for different, possibly incompatible, notions of defeasibility in subsumption resulting from the fact that a given object may be more exceptional than another in some context but less exceptional in another. Defeasibility therefore introduces a new facet of contextual reasoning not present in *deductive* reasoning. In recent work [21] we addressed this limitation by allowing different orderings on objects, using preference relations on role interpretations [17]. Here we complete the picture by also providing a proof-theoretic counterpart and associated results.

After setting up the notation and our access-control example in Section 2, we give a summary of our context-based defeasible DL (Section 3). In Section 4, we define a tableau-based algorithm for checking consistency of contextual defeasible knowledge bases. The paper concludes with a discussion on future directions of investigation.

2 Notation and an example

We assume finite and pairwise disjoint sets C, R and I standing for, respectively, concept, role and individual names. With $A, B, \ldots$ we denote concept names, with $r, s, \ldots$, role names, and with $a, b, \ldots$, individual names. In the access-control scenario above we could have, for example, $\mathsf{C} = \{\mathsf{Classified}, \mathsf{Employee}, \mathsf{Graduate}, \mathsf{Intern}, \mathsf{ResAssoc}\}$, $\mathsf{R} = \{\mathsf{hasAcc}, \mathsf{hasJob}, \mathsf{hasQua}\}$, and $\mathsf{I} = \{\mathsf{anne}, \mathsf{bill}, \mathsf{chris}, \mathsf{doc123}\}$, with the respective obvious intuitions. Complex concepts are denoted $C, D, \ldots$

Figure 1 depicts an interpretation for our access-control example with domain $\Delta^{\mathcal{I}} = \{x_i \mid 0 \leq i \leq 11\}$, and interpreting the elements of the vocabulary as follows: $\mathsf{Classified}^{\mathcal{I}} = \{x_{10}\}$, $\mathsf{Employee}^{\mathcal{I}} = \{x_0, x_4, x_5, x_9\}$, $\mathsf{Graduate}^{\mathcal{I}} = \{x_4, x_5, x_6, x_9\}$, $\mathsf{Intern}^{\mathcal{I}} = \{x_0, x_4\}$, $\mathsf{ResAssoc}^{\mathcal{I}} = \{x_5, x_6, x_7\}$, $\mathsf{hasAcc}^{\mathcal{I}} = \{(x_4, x_{10}), (x_9, x_{10}), (x_6, x_{10}), (x_6, x_{11})\}$, $\mathsf{hasJob}^{\mathcal{I}} = \{(x_0, x_3), (x_4, x_3), (x_9, x_3), (x_5, x_1), (x_6, x_1)\}$, and $\mathsf{hasQua}^{\mathcal{I}} = \{(x_4, x_8), (x_9, x_8), (x_5, x_2), (x_6, x_2), (x_7, x_2)\}$. Further, $\mathsf{anne}^{\mathcal{I}} = x_5$, $\mathsf{bill}^{\mathcal{I}} = x_0$, $\mathsf{chris}^{\mathcal{I}} = x_6$, and $\mathsf{doc123}^{\mathcal{I}} = x_{10}$.

The knowledge base $\mathcal{KB} = \mathcal{T} \cup \mathcal{A}$, with $\mathcal{T}$ and $\mathcal{A}$ as below, is a first stab at formalising our access-control example:

$$\mathcal{T} = \left\{ \begin{array}{c} \mathsf{Intern} \sqsubseteq \mathsf{Employee}, \\ \mathsf{Employee} \sqsubseteq \exists \mathsf{hasJob}.\top, \\ \mathsf{Graduate} \sqsubseteq \mathsf{hasQua}.\top, \\ \mathsf{Employee} \sqsubseteq \exists \mathsf{hasAcc}.\mathsf{Classified}, \\ \mathsf{Intern} \sqsubseteq \neg\exists \mathsf{hasAcc}.\mathsf{Classified}, \\ \mathsf{Intern} \sqcap \mathsf{Graduate} \sqsubseteq \exists \mathsf{hasAcc}.\mathsf{Classified}, \\ \mathsf{ResAssoc} \sqsubseteq \neg \mathsf{Employee}, \\ \mathsf{ResAssoc} \sqsubseteq \mathsf{Graduate} \end{array} \right\} \mathcal{A} = \left\{ \begin{array}{c} \mathsf{anne} : \mathsf{ResAssoc}, \\ \mathsf{chris} : \mathsf{ResAssoc}, \\ \mathsf{doc123} : \mathsf{Classified}, \\ (\mathsf{chris}, \mathsf{doc123}) : \mathsf{hasAcc} \end{array} \right\}$$

It is not hard to see that this knowledge base is satisfiable and to check that $\mathcal{KB} \models \mathsf{Intern} \sqsubseteq \bot$. Incoherence of the knowledge base is but one of the (many) reasons to go defeasible. Armed with a notion of *defeasible subsumption* of the form $C \mathrel{\sqsubset\!\!\!\sim} D$ [15], of which the intuition is "normally, C is subsumed by D", formalised by the adoption of

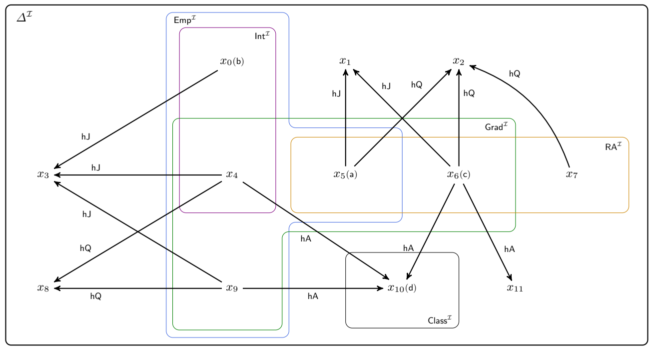

Fig. 1. An $\mathcal{ALC}$ interpretation for C, R and I as above. For the sake of presentation, concept, role and individual names have been abbreviated.

a preferential semantics *à la* Shoham [41], we can give a more refined formalisation of our scenario example with $\mathcal{KB} = \mathcal{T} \cup \mathcal{D} \cup \mathcal{A}$, where $\mathcal{T}$ and $\mathcal{D}$ are given below ($\mathcal{D}$ standing for a *defeasible TBox*) and $\mathcal{A}$ is as above:

$$\mathcal{T} = \left\{ \begin{array}{c} \mathsf{Intern} \sqsubseteq \mathsf{Employee}, \\ \mathsf{Employee} \sqsubseteq \exists \mathsf{hasJob}.\top, \\ \mathsf{Graduate} \sqsubseteq \mathsf{hasQua}.\top \end{array} \right\} \mathcal{D} = \left\{ \begin{array}{c} \mathsf{Employee} \precsim \exists \mathsf{hasAcc}.\mathsf{Classified}, \\ \mathsf{Intern} \precsim \neg\exists \mathsf{hasAcc}.\mathsf{Classified}, \\ \mathsf{Intern} \sqcap \mathsf{Graduate} \precsim \exists \mathsf{hasAcc}.\mathsf{Classified}, \\ \mathsf{ResAssoc} \precsim \neg \mathsf{Employee}, \\ \mathsf{ResAssoc} \precsim \mathsf{Graduate} \end{array} \right\}$$

Then, one could ask whether intern research associates are usually graduates, and whether they should usually have access to classified information. It soon becomes clear that modelling defeasible information is more challenging than modelling classical information, and that it becomes problematic when defeasible information relating to different contexts are not modelled independently.

Suppose, for example, that Chris is a graduate research associate who is also an employee, and Anne is a research associate who is neither a graduate nor an employee. In any preferential model of the defeasible $\mathcal{KB}$, both Chris and Anne are exceptional in the class of research associates. This follows because Chris is an exceptional research associate w.r.t. employment status, and Anne is an exceptional research associate w.r.t. qualification. Also, in any preferential model of $\mathcal{KB}$ Chris and Anne are either incomparable, or one of them is more normal than the other. Since context has not been taken into account, there is no model in which Anne is more normal than Chris w.r.t. employment, but Chris is more normal than Anne w.r.t. qualification.

3 Contextual defeasible $\mathcal{ALC}$

Contextual defeasible $\mathcal{ALC}$ ($d\mathcal{ALC}$) smoothly combines in a single logical framework the following features: all classical $\mathcal{ALC}$ constructs; defeasible value and existential restrictions [12, 17]; defeasible concept inclusions [15], and context [18, 21].

Let C, R and I be as before. Complex $d\mathcal{ALC}$ concepts are denoted $C, D, \ldots$, and are built according to the rules:

$$C ::= \top \mid \bot \mid \mathsf{C} \mid (\neg C) \mid (C \sqcap C) \mid (C \sqcup C) \mid (\exists r.C) \mid (\forall r.C) \mid (\exists\!\!\!\sim r.C) \mid (\forall\!\!\!\sim r.C)$$

With $\mathcal{L}_{d\mathcal{ALC}}$ we denote the language of all $d\mathcal{ALC}$ concepts (including all $\mathcal{ALC}$ concepts). Again, when writing down elements of $\mathcal{L}_{d\mathcal{ALC}}$, we shall omit parentheses whenever they are not necessary for disambiguation. An example of $d\mathcal{ALC}$ concept is $\mathsf{ResAssoc} \sqcap (\forall\!\!\!\sim \mathsf{hasAcc}.\neg\mathsf{Classified}) \sqcap (\exists\mathsf{hasAcc}.\mathsf{Classified})$, denoting those research associates whose normal access is only to non-classified info but who also turn out to have some (exceptional) access to a classified document.

The semantics of $d\mathcal{ALC}$ is anchored in the well-known preferential approach to non-monotonic reasoning [34, 35, 41] and its extensions [8–11, 16, 19, 20], especially those in DLs [15, 17, 28, 38, 43].

Let X be a set. With $\#X$ we denote the *cardinality* of X. A binary relation is a *strict partial order* if it is irreflexive and transitive. If $<$ is a strict partial order on X, with $\min_< X =_{\text{def}} \{x \in X \mid \text{there is no } y \in X \text{ s.t. } y < x\}$ we denote the *minimal elements* of X w.r.t. $<$. A strict partial order on a set X is *well-founded* if for every $\emptyset \neq X' \subseteq X$, $\min_< X' \neq \emptyset$.

Definition 1 (Ordered interpretation). *An **ordered interpretation** is a tuple $\mathcal{O} =_{\text{def}} \langle \Delta^{\mathcal{O}}, \cdot^{\mathcal{O}}, \ll^{\mathcal{O}} \rangle$ such that:*

- *$\langle \Delta^{\mathcal{O}}, \cdot^{\mathcal{O}} \rangle$ is an $\mathcal{ALC}$ interpretation, with $A^{\mathcal{O}} \subseteq \Delta^{\mathcal{O}}$, for each $A \in \mathsf{C}$, $r^{\mathcal{O}} \subseteq \Delta^{\mathcal{O}} \times \Delta^{\mathcal{O}}$, for each $r \in \mathsf{R}$, and $a^{\mathcal{O}} \in \Delta^{\mathcal{O}}$, for each $a \in \mathsf{I}$, and*
- *$\ll^{\mathcal{O}} =_{\text{def}} \langle \ll^{\mathcal{O}}_{r_1}, \ldots, \ll^{\mathcal{O}}_{r_{\#\mathsf{R}}} \rangle$, where $\ll^{\mathcal{O}}_{r_i} \subseteq r_i^{\mathcal{O}} \times r_i^{\mathcal{O}}$, for $i = 1, \ldots, \#\mathsf{R}$, and such that each $\ll^{\mathcal{O}}_{r_i}$ is a well-founded strict partial order.*

Given $\mathcal{O} = \langle \Delta^{\mathcal{O}}, \cdot^{\mathcal{O}}, \ll^{\mathcal{O}} \rangle$, the intuition of $\Delta^{\mathcal{O}}$ and $\cdot^{\mathcal{O}}$ is the same as in a standard $\mathcal{ALC}$ interpretation. The intuition underlying each of the orderings in $\ll^{\mathcal{O}}$ is that they play the role of *preference relations* (or *normality orderings*), in a sense similar to the preference orders introduced by Shoham [41] in a propositional setting, and investigated by Kraus et al. [34, 35] and others [9–11, 13, 26]: The pairs (x, y) that are lower down in the ordering $\ll^{\mathcal{O}}_{r_i}$ are deemed as most normal (or typical, or expected, or conventional) in the context of (the interpretation of) r_i.

Figure 2 depicts an ordered interpretation in our example, where $\Delta^{\mathcal{O}}$ and $\cdot^{\mathcal{O}}$ are as in the interpretation $\mathcal{I}$ shown in Figure 1, and $\ll^{\mathcal{O}} = \langle \ll^{\mathcal{O}}_{\mathsf{hasAcc}}, \ll^{\mathcal{O}}_{\mathsf{hasJob}}, \ll^{\mathcal{O}}_{\mathsf{hasQua}} \rangle$, where $\ll^{\mathcal{O}}_{\mathsf{hasAcc}} = \{(x_6x_{11}, x_6x_{10})\}$, $\ll^{\mathcal{O}}_{\mathsf{hasJob}} = \{(x_9x_3, x_0x_3), (x_0x_3, x_4x_3), (x_9x_3, x_4x_3), (x_0x_3, x_5x_1), (x_9x_3, x_5x_1), (x_6x_1, x_5x_1)\}$, and $\ll^{\mathcal{O}}_{\mathsf{hasQua}} = \{(x_5x_2, x_6x_2), (x_6x_2, x_7x_2), (x_5x_2, x_7x_2)\}$. (For the sake of readability, we shall henceforth sometimes write r-tuples of the form (x, y) as xy.)

In the following definition we extend ordered interpretations to complex concepts of the language.

Definition 2 (Interpretation of concepts). *Let $\mathcal{O} = \langle \Delta^{\mathcal{O}}, \cdot^{\mathcal{O}}, \ll^{\mathcal{O}} \rangle$, let $r \in \mathsf{R}$ and, for each $x \in \Delta^{\mathcal{O}}$, let $r^{\mathcal{O}|x} =_{\text{def}} r^{\mathcal{O}} \cap (\{x\} \times \Delta^{\mathcal{O}})$ (i.e., the restriction of the domain of $r^{\mathcal{O}}$*

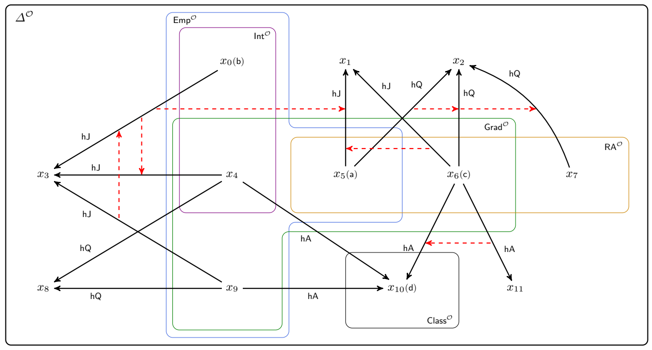

Fig. 2. An ordered interpretation. For the sake of presentation, we omit the transitive $\ll_r^{\mathcal{O}}$-arrows.

to $\{x\}$). The interpretation function $\cdot^{\mathcal{O}}$ interprets $d\mathcal{ALC}$ concepts as follows:

$$\top^{\mathcal{O}} =_{\text{def}} \Delta^{\mathcal{O}}; \quad \bot^{\mathcal{O}} =_{\text{def}} \emptyset; \quad (\neg C)^{\mathcal{O}} =_{\text{def}} \Delta^{\mathcal{O}} \setminus C^{\mathcal{O}};$$
$$(C \sqcap D)^{\mathcal{O}} =_{\text{def}} C^{\mathcal{O}} \cap D^{\mathcal{O}}; \quad (C \sqcup D)^{\mathcal{O}} =_{\text{def}} C^{\mathcal{O}} \cup D^{\mathcal{O}};$$
$$(\exists r.C)^{\mathcal{O}} =_{\text{def}} \{x \in \Delta^{\mathcal{O}} \mid r^{\mathcal{O}}(x) \cap C^{\mathcal{O}} \neq \emptyset\}; \ (\forall r.C)^{\mathcal{O}} =_{\text{def}} \{x \in \Delta^{\mathcal{O}} \mid r^{\mathcal{O}}(x) \subseteq C^{\mathcal{O}}\};$$
$$(\widetilde{\exists} r.C)^{\mathcal{O}} =_{\text{def}} \{x \in \Delta^{\mathcal{O}} \mid \min_{\ll_r^{\mathcal{O}}}(r^{\mathcal{O}|x})(x) \cap C^{\mathcal{O}} \neq \emptyset\};$$
$$(\widetilde{\forall} r.C)^{\mathcal{O}} =_{\text{def}} \{x \in \Delta^{\mathcal{O}} \mid \min_{\ll_r^{\mathcal{O}}}(r^{\mathcal{O}|x})(x) \subseteq C^{\mathcal{O}}\}.$$

Analogously to the classical case, $\widetilde{\forall}$ and $\widetilde{\exists}$ are dual to each other. As an example, in the ordered interpretation $\mathcal{O}$ of Figure 2, we have that $(\widetilde{\exists}\mathsf{hasAcc}.\mathsf{Classified})^{\mathcal{O}} = \emptyset = (\neg\, \widetilde{\forall}\mathsf{hasAcc}.\neg\mathsf{Classified})^{\mathcal{O}}$, whereas $(\exists\mathsf{hasAcc}.\mathsf{Classified})^{\mathcal{O}} = \{x_6\}$.

Defeasible $\mathcal{ALC}$ also adds *contextual* defeasible subsumption statements to knowledge bases. Given $C, D \in \mathcal{L}_{d\mathcal{ALC}}$ and $r \in \mathsf{R}$, a statement of the form $C \mathrel{\sqsubset\!\!\!\sim}_r D$ is a (contextual) *defeasible concept inclusion* (DCI), read "C is usually subsumed by D in the context r". A $d\mathcal{ALC}$ *defeasible TBox* $\mathcal{D}$ (or dTBox $\mathcal{D}$ for short) is a finite set of DCIs. A $d\mathcal{ALC}$ *classical TBox* $\mathcal{T}$ (or TBox $\mathcal{T}$ for short) is a finite set of (classical) subsumption statements $C \sqsubseteq D$ (i.e., $\mathcal{T}$ may contain defeasible concept constructs, but not defeasible concept inclusions). Given $\mathcal{T}$, $\mathcal{D}$ and $\mathcal{A}$, with $\mathcal{KB} =_{\text{def}} \mathcal{T} \cup \mathcal{D} \cup \mathcal{A}$ we denote a $d\mathcal{ALC}$ *knowledge base*, a.k.a. a *defeasible ontology*, an example of which is given below:

$$\mathcal{T} = \left\{ \begin{array}{c} \mathsf{Intern} \sqsubseteq \mathsf{Employee}, \\ \mathsf{Employee} \sqsubseteq \exists\mathsf{hasJob}.\top, \\ \mathsf{Graduate} \sqsubseteq \mathsf{hasQua}.\top, \\ \mathsf{ResAssoc} \sqsubseteq \widetilde{\forall}\mathsf{hasAcc}.\neg\mathsf{Classified} \end{array} \right\} \quad \mathcal{A} = \left\{ \begin{array}{c} \mathsf{anne} : \mathsf{Employee}, \\ \mathsf{anne} : \mathsf{ResAssoc}, \\ \mathsf{bill} : \mathsf{Intern}, \\ \mathsf{chris} : \mathsf{ResAssoc}, \\ \mathsf{doc123} : \mathsf{Classified}, \\ (\mathsf{chris}, \mathsf{doc123}) : \mathsf{hasAcc} \end{array} \right\}$$

$$\mathcal{D} = \left\{ \begin{array}{c} \mathsf{Employee} \precsim_{\mathsf{hasJob}} \exists \mathsf{hasAcc}.\mathsf{Classified}, \\ \mathsf{Intern} \precsim_{\mathsf{hasJob}} \neg\exists \mathsf{hasAcc}.\mathsf{Classified}, \\ \mathsf{Intern} \sqcap \mathsf{Graduate} \precsim_{\mathsf{hasJob}} \exists \mathsf{hasAcc}.\mathsf{Classified}, \\ \mathsf{ResAssoc} \precsim_{\mathsf{hasJob}} \neg \mathsf{Employee}, \\ \mathsf{ResAssoc} \precsim_{\mathsf{hasQua}} \mathsf{Graduate} \end{array} \right\}$$

Definition 3 (Satisfaction). *Let $\mathcal{O} = \langle \Delta^{\mathcal{O}}, \cdot^{\mathcal{O}}, \ll^{\mathcal{O}} \rangle$, $r \in \mathsf{R}$, $C, D \in \mathcal{L}_{d\mathcal{ALC}}$, and $a, b \in \mathsf{I}$. Define $\prec_r^{\mathcal{O}} \subseteq \Delta^{\mathcal{O}} \times \Delta^{\mathcal{O}}$ as follows:*

$$\prec_r^{\mathcal{O}} =_{\text{def}} \{(x, y) \mid \textit{there is } (x, z) \in r^{\mathcal{O}} \textit{ s.t. for all } (y, v) \in r^{\mathcal{O}}, ((x, z), (y, v)) \in \ll_r^{\mathcal{O}}\}.$$

*The **satisfaction relation** $\Vdash$ is defined as follows:*

$$\mathcal{O} \Vdash C \sqsubseteq D \quad \textit{if} \quad C^{\mathcal{O}} \subseteq D^{\mathcal{O}}; \qquad \mathcal{O} \Vdash C \precsim_r D \quad \textit{if} \quad \min_{\prec_r^{\mathcal{O}}} C^{\mathcal{O}} \subseteq D^{\mathcal{O}};$$
$$\mathcal{O} \Vdash a : C \quad \textit{if} \quad a^{\mathcal{O}} \in C^{\mathcal{O}}; \qquad \mathcal{O} \Vdash (a, b) : r \quad \textit{if} \quad (a^{\mathcal{O}}, b^{\mathcal{O}}) \in r^{\mathcal{O}}.$$

*If $\mathcal{O} \Vdash \alpha$, then we say $\mathcal{O}$ **satisfies** α. $\mathcal{O}$ satisfies a $d\mathcal{ALC}$ knowledge base $\mathcal{KB}$, written $\mathcal{O} \Vdash \mathcal{KB}$, if $\mathcal{O} \Vdash \alpha$ for every $\alpha \in \mathcal{KB}$, in which case we say $\mathcal{O}$ is a **model** of $\mathcal{KB}$. We say $\mathcal{KB}$ is **preferentially consistent** if it admits a model. We say $C \in \mathcal{L}_{d\mathcal{ALC}}$ (resp. $r \in \mathsf{R}$) is **satisfiable** w.r.t. $\mathcal{KB}$ if there is a model $\mathcal{O}$ of $\mathcal{KB}$ s.t. $C^{\mathcal{O}} \neq \emptyset$ (resp. $r^{\mathcal{O}} \neq \emptyset$).*

One can check that the interpretation $\mathcal{O}$ in Figure 2 satisfies the above knowledge base. To help in seeing why, Figure 3 depicts the contextual orderings on objects (represented with dotted arrows) induced from those on roles in $\mathcal{O}$ as specified in Definition 3.

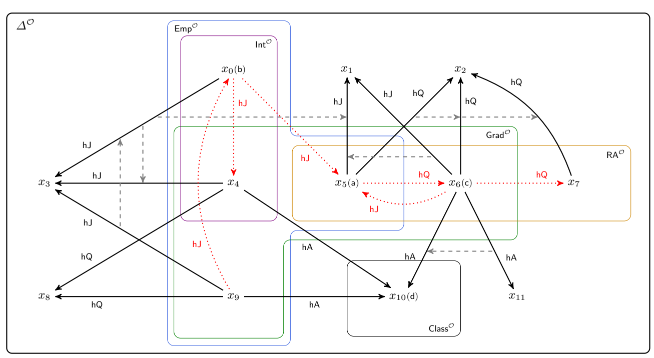

Fig. 3. Induced orderings on objects from the role orderings in Figure 2. For the sake of presentation, we omit the transitive $\prec_r^{\mathcal{O}}$-arrows.

It follows from Definition 3 that, if $\ll_r^{\mathcal{O}} = \emptyset$, i.e., if no r-tuple is preferred to another, then $\precsim_r$ reverts to a context-agnostic classical $\sqsubseteq$. A similar observation holds for individual concept inclusions: if $(C \sqcap \exists r.\top)^{\mathcal{O}} = \emptyset$, then $C \precsim_r D$ reverts to $C \sqsubseteq D$. This reflects the intuition that the context r is taken into account through the preference order on $r^{\mathcal{O}}$. In the absence of any preference, the context becomes irrelevant. This also shows why the classical counterpart of $\sqsubsetapprox_r$ is independent of r — context is taken into account in the form of a preference order, but preference has no bearing on the semantics of $\sqsubseteq$.

Contextual defeasible subsumption $\sqsubsetapprox_r$ can also be viewed as defeasible subsumption based on a preference order on objects in the domain of $r^{\mathcal{O}}$ obtained from $\ll_r^{\mathcal{O}}$. Non-contextual defeasible subsumption can then be obtained as a special case by introducing a new role name r and axiom $\top \sqsubseteq \exists r.\top$.

Given a $d\mathcal{ALC}$ knowledge base $\mathcal{KB}$, a fundamental task from the standpoint of knowledge representation and reasoning is that of deciding which statements follow from $\mathcal{KB}$ and which do not.

Definition 4 (Preferential entailment). *A statement α is **preferentially entailed** by a $d\mathcal{ALC}$ knowledge base $\mathcal{KB}$, written $\mathcal{KB} \models_{\mathsf{pref}} \alpha$, if $\mathcal{O} \Vdash \alpha$ for every $\mathcal{O}$ s.t. $\mathcal{O} \Vdash \mathcal{KB}$.*

The following lemma shows that deciding preferential entailment of GCIs and assertions can be reduced to $d\mathcal{ALC}$ knowledge base satisfiability, a result that will be used in the definition of a tableau system in Section 4. Its proof is analogous to that of its classical counterpart in the DL literature and we shall omit it here:

Lemma 1. *Let $\mathcal{KB}$ be a $d\mathcal{ALC}$ knowledge base and let a be an individual name not occurring in $\mathcal{KB}$. For every $C, D \in \mathcal{L}_{d\mathcal{ALC}}$, $\mathcal{KB} \models C \sqsubseteq D$ iff $\mathcal{KB} \models C \sqcap \neg D \sqsubseteq \bot$ iff $\mathcal{KB} \cup \{a : C \sqcap \neg D\}$ is unsatisfiable. Moreover, for every $b \in \mathsf{I}$ and every $C \in \mathcal{L}_{d\mathcal{ALC}}$, $\mathcal{KB} \models b : C$ iff $\mathcal{KB} \cup \{b : \neg C\}$ is unsatisfiable.*

It turns out that deciding preferential entailment of DCIs too can be reduced to $d\mathcal{ALC}$ knowledge base satisfiability, but first, we introduce the tableau-based algorithm for deciding preferential consistency.

4 Tableau for preferential reasoning in $d\mathcal{ALC}$

In this section, we define a tableau method for deciding preferential consistency of a $d\mathcal{ALC}$ knowledge base. Our terminology and presentation follow those of Baader et al. [4] in the classical case.

We start by observing that, for every ordered interpretation $\mathcal{O}$ and every $C, D \in \mathcal{L}_{d\mathcal{ALC}}$, $\mathcal{O} \Vdash C \sqsubseteq D$ if and only if $\mathcal{O} \Vdash \top \sqsubseteq \neg C \sqcup D$. In that respect, we can assume w.l.o.g. that all GCIs in a TBox are of the form $\top \sqsubseteq E$, for some $E \in \mathcal{L}_{d\mathcal{ALC}}$.

Note also that we can assume w.l.o.g. that the ABox is not empty, for if it is, one can add to it the trivial assertion $a : \top$, for some new individual name a. It is easy to see that the resulting (non-empty) ABox is preferentially equivalent to the original one.

Definition 5 (Subconcepts). *Let $C \in \mathcal{L}_{d\mathcal{ALC}}$. The set of **subconcepts** of C, denoted $\mathsf{sub}(C)$, is defined inductively as follows:*

- *If $C = A$, for $A \in \mathsf{C} \cup \{\top, \bot\}$, then $\mathsf{sub}(C) =_{\text{def}} \{A\}$;*
- *If $C = C_1 \sqcap C_2$ or $C = C_1 \sqcup C_2$, then $\mathsf{sub}(C) =_{\text{def}} \{C\} \cup \mathsf{sub}(C_1) \cup \mathsf{sub}(C_2)$;*

– *If* $C = \neg D$ *or* $C = \exists r.D$ *or* $C = \forall r.D$ *or* $C = \tilde{\exists} r.D$ *or* $C = \tilde{\forall} r.D$*, then* $\mathsf{sub}(C) =_{\text{def}} \{C\} \cup \mathsf{sub}(D)$*.*

Given a knowledge base $\mathcal{KB} = \mathcal{T} \cup \mathcal{D} \cup \mathcal{A}$*, the set of subconcepts of* $\mathcal{KB}$ *is defined as* $\mathsf{sub}(\mathcal{KB}) =_{\text{def}} \mathsf{sub}(\mathcal{T}) \cup \mathsf{sub}(\mathcal{D}) \cup \mathsf{sub}(\mathcal{A})$*, where*

$$\mathsf{sub}(\mathcal{T}) =_{\text{def}} \bigcup_{C \sqsubseteq D \in \mathcal{T}} (\mathsf{sub}(C) \cup \mathsf{sub}(D)) \qquad \mathsf{sub}(\mathcal{A}) =_{\text{def}} \bigcup_{a:C \in \mathcal{A}} \mathsf{sub}(C)$$

$$\mathsf{sub}(\mathcal{D}) =_{\text{def}} \bigcup_{C \underset{\sim}{\sqsubset}_r D \in \mathcal{D}} (\mathsf{sub}(C) \cup \mathsf{sub}(D))$$

We say that an individual name a *appears* in an ABox $\mathcal{A}$ if $\mathcal{A}$ contains an assertion of the form $a : C$, $(a, b) : r$ or $(b, a) : r$, for some $C \in \mathcal{L}_{d\mathcal{ALC}}$, $r \in \mathsf{R}$ and $b \in \mathsf{I}$.

Definition 6 **(a-concepts).** *Let* $\mathcal{A}$ *be an ABox and let* a *be an individual name appearing in* $\mathcal{A}$*. With* $\mathsf{con}_{\mathcal{A}}(a) =_{\text{def}} \{C \mid a : C \in \mathcal{A}\}$ *we denote the* ***set of concepts that*** a ***is an instance of*** *w.r.t.* $\mathcal{A}$*.*

We are now ready for the definition of the expansion rules for $d\mathcal{ALC}$-concepts. They are shown in Figure 4. The $\sqcap$-, $\sqcup$-, $\forall$-, and $\mathcal{T}$-rules work as in the classical case [4], whereas the remaining rules handle the additional $d\mathcal{ALC}$ constructs according to our preferential semantics. We shall explain them in more detail below. Before doing so, we need a few more definitions, in particular of what it means for an individual to be *blocked*, as tested by the $\exists$-, $\tilde{\exists}$-, and $\underset{\sim}{\sqsubset}$-rules and needed to ensure termination of the algorithm we shall present.

As can be seen in the expansion rules, our tableau method makes use of a few auxiliary structures, which are built incrementally during the search for a model of the input knowledge base. The first one is a partial order on pairs of individuals $\rho^r_{\mathcal{A}}$, for each $r \in \mathsf{R}$. Its purpose is to build the skeleton of an r-preference relation on pairs of individual names appearing in an ABox $\mathcal{A}$. In the unravelling of the complete clash-free ABox (see below), if there is any, $\rho^r_{\mathcal{A}}$ is used to define a preference relation on the interpretation of role r in the constructed ordered interpretation.

The second auxiliary structure is a pre-order $\sigma^r_{\mathcal{A}}$ on individual names, for each $r \in \mathsf{R}$. It fits the purpose of keeping track of which individuals are to be seen as more normal (or typical) relative to others in the application of the $\underset{\sim}{\sqsubset}$-rule (see Figure 4) so that the associated $\rho^r_{\mathcal{A}}$-ordering can be completed (by the $\ll$-rule) and, in the unravelling of the model, deliver an induced $\prec_r$ that is faithful to $\sigma^r_{\mathcal{A}}$. (This point will be made more clearly in the explanation of the relevant rules. In particular, the reason why $\sigma^r_{\mathcal{A}}$ is a pre-order and not a partial order like $\rho^r_{\mathcal{A}}$ will be explained in the soundness proof.) Intuitively, $\sigma^r_{\mathcal{A}}$ corresponds to the converse of the preference order introduced in Definition 3.

Finally, the third structure used in the expansion rules is a labelling function $\tau^r_{\mathcal{A}}(a)$ mapping an individual name a to the set of concepts a ought to be a minimal instance of in the context r w.r.t. the ABox $\mathcal{A}$. The purpose of $\tau^r_{\mathcal{A}}(a)$ is threefold: (*i*) it is needed to ensure the minimal elements of a concept C inherit all defeasible properties encoded in the DCIs (see $\underset{\sim}{\sqsubset}$-rule); (*ii*) it flags that every individual more preferred than a should be marked as $\neg C$, as performed by the min-rule, and (*iii*) it plays a role in the blocking condition (see below) to prevent the generation of an infinite chain of increasingly more normal elements in $\sigma^r_{\mathcal{A}}$. Note that $\rho^r_{\mathcal{A}}$, $\sigma^r_{\mathcal{A}}$ and $\tau^r_{\mathcal{A}}(a)$ are only used in the inner workings of the tableau and are not accessible to the user.

Definition 7 **(r-ancestor).** *Let $\mathcal{A}$ be an ABox, $a, b \in \mathsf{I}$, and $r \in \mathsf{R}$. If $(a, b) : r \in \mathcal{A}$, we say b is an r-**successor** of a and a is an r-**predecessor** of b. The transitive closure of the r-predecessor (resp. r-successor) relation is called r-**ancestor** (resp. r-**descendant**).*

Definition 8 **($\sigma^r_{\mathcal{A}}$-ancestor).** *Let $\mathcal{A}$ be an ABox, $a, b \in \mathsf{I}$, and $r \in \mathsf{R}$. If $(a, b) \in \sigma^r_{\mathcal{A}}$, we say b is a $\sigma^r_{\mathcal{A}}$-**successor** of a and a is an $\sigma^r_{\mathcal{A}}$-**predecessor** of b. The transitive closure of the $\sigma^r_{\mathcal{A}}$-predecessor (resp. $\sigma^r_{\mathcal{A}}$-successor) relation is called $\sigma^r_{\mathcal{A}}$-**ancestor** (resp. $\sigma^r_{\mathcal{A}}$-**descendant**).*

An individual is called a **root** if it has neither an r-ancestor nor a $\sigma^r_{\mathcal{A}}$-ancestor.

$\sqcap$-**rule**:	**if**	1. $a : C \sqcap D \in \mathcal{A}$, and
		2. $\{a : C, a : D\} \not\subseteq \mathcal{A}$
	then	$\mathcal{A} := \mathcal{A} \cup \{a : C, a : D\}$
$\sqcup$-**rule**:	**if**	1. $a : C \sqcup D \in \mathcal{A}$, and
		2. $\{a : C, a : D\} \cap \mathcal{A} = \emptyset$
	then	$\mathcal{A} := \mathcal{A} \cup \{a : E\}$, for some $E \in \{C, D\}$
$\exists$-**rule**:	**if**	1. $a : \exists r.C \in \mathcal{A}$, and
		2. there is no b s.t. $\{(a, b) : r, b : C\} \subseteq \mathcal{A}$, and
		3. a is not blocked
	then	(a) $\mathcal{A} := \mathcal{A} \cup \{(a, c) : r, c : C\}$, for c new in $\mathcal{A}$, **or**
		(b) $\mathcal{A} := \mathcal{A} \cup \{(a, c) : r, c : C, (a, d) : r\}$, for c, d new in $\mathcal{A}$, and $\rho^r_{\mathcal{A}} := \rho^r_{\mathcal{A}} \cup \{(ad, ac)\}$
$\forall$-**rule**:	**if**	1. $\{a : \forall r.C, (a, b) : r\} \subseteq \mathcal{A}$, and
		2. $b : C \notin \mathcal{A}$
	then	$\mathcal{A} := \mathcal{A} \cup \{b : C\}$
$\tilde{\exists}$-**rule**:	**if**	1. $a : \tilde{\exists} r.C \in \mathcal{A}$, and
		2. there is no b s.t. (*i*) $\{(a, b) : r, b : C\} \subseteq \mathcal{A}$, and (*ii*) there is no c s.t. $(ac, ab) \in \rho^r_{\mathcal{A}}$, and
		3. a is not blocked
	then	$\mathcal{A} := \mathcal{A} \cup \{(a, d) : r, d : C\}$, for d new in $\mathcal{A}$
$\tilde{\forall}$-**rule**:	**if**	1. $\{a : \tilde{\forall} r.C, (a, b) : r\} \subseteq \mathcal{A}$, and
		2. there is no c s.t. $(ac, ab) \in \rho^r_{\mathcal{A}}$, and
		3. $b : C \notin \mathcal{A}$
	then	$\mathcal{A} := \mathcal{A} \cup \{b : C\}$
$\mathcal{T}$-**rule**:	**if**	1. a appears in $\mathcal{A}$, $\top \sqsubseteq D \in \mathcal{T}$, and
		2. $a : D \notin \mathcal{A}$
	then	$\mathcal{A} := \mathcal{A} \cup \{a : D\}$
$\tilde{\sqsubseteq}$-**rule**:	**if**	1. a appears in $\mathcal{A}$, $C \mathrel{\tilde{\sqsubseteq}}_r D \in \mathcal{D}$, and
		2. $\{a : \neg C, a : D\} \cap \mathcal{A} = \emptyset$, and
		3. either $a : C \notin \mathcal{A}$ or there is no b s.t. $b : C \in \mathcal{A}$ and $(a, b) \in \sigma^r_{\mathcal{A}}$, and
		4. a is not blocked
	then	(a) $\mathcal{A} := \mathcal{A} \cup \{a : \neg C\}$, **or**
		(b) $\mathcal{A} := \mathcal{A} \cup \{a : C, c : C, c : D\}$, for c new in $\mathcal{A}$, $\sigma^r_{\mathcal{A}} := \sigma^r_{\mathcal{A}} \cup \{(a, c)\}$, and $\tau^r_{\mathcal{A}}(c) := \{C\}$ **or**
		(c) $\mathcal{A} := \mathcal{A} \cup \{a : D\}$
min-**rule**:	**if**	1. $C \in \tau^r_{\mathcal{A}}(a)$, and
		2. $b : \neg C \notin \mathcal{A}$, for some b s.t. $(a, b) \in (\sigma^r_{\mathcal{A}})^+$
	then	$\mathcal{A} := \mathcal{A} \cup \{b : \neg C\}$
$\ll$-**rule**:	**if**	1. $(b, a) \in \sigma^r_{\mathcal{A}}$, and
		2. there is no c s.t. $(ac, bd) \in \rho^r_{\mathcal{A}}$ for every $(b, d) : r \in \mathcal{A}$, and
		3. a is not blocked
	then	$\mathcal{A} := \mathcal{A} \cup \{(a, e) : r\}$, for e new in $\mathcal{A}$, and $\rho^r_{\mathcal{A}} := \rho^r_{\mathcal{A}} \cup \{(ae, bf) \mid (b, f) : r \in \mathcal{A}\}$

Fig. 4. Expansion rules for the $d\mathcal{ALC}$ tableau.

The following definition is used in the expansion rules of Figure 4 to ensure termination:

Definition 9 (Blocking). *Let $\mathcal{A}$ be an ABox, $a, b \in \mathsf{I}$, and let $\sigma^r_{\mathcal{A}}$ and $\tau^r_{\mathcal{A}}$ be as above. We say that b is **blocked** by a in $\mathcal{A}$ in the context r if (1) a is either an r-ancestor or a $\sigma^r_{\mathcal{A}}$-ancestor of b, (2) $\mathsf{con}_{\mathcal{A}}(b) \subseteq \mathsf{con}_{\mathcal{A}}(a)$, and (3) $\tau^r_{\mathcal{A}}(b) \subseteq \tau^r_{\mathcal{A}}(a)$. We say b is blocked in $\mathcal{A}$ if itself or some r-ancestor or $\sigma^r_{\mathcal{A}}$-ancestor of b is blocked by some individual.*

The $\sqcap$-, $\sqcup$-, $\forall$-, and $\mathcal{T}$-rules in Figure 4 are as in the classical case and need no further explanation.

The $\exists\!\!\!\sim$-rule creates a most preferred (relative to individual a) r-link to a new individual falling under concept C. Notice that this is achieved by just adding an assertion $(a, d) : r$ to $\mathcal{A}$, for d new in $\mathcal{A}$, since there shall never be (a, e) with $(ae, ad) \in \rho^r_{\mathcal{A}}$.

The $\forall\!\!\!\sim$-rule is analogous to the $\forall$-rule, but propagates a concept C only to those individuals across preferred r-links (i.e., r-links that are minimal in $\rho^r_{\mathcal{A}}$).

The $\exists$-rule handles the creation of an r-successor without the information whether such an r-link is relatively preferred or not. In this case, both possibilities have to be explored, which is formalised by the or-branching in the rule. In one case, a preferred r-link is created just as in the $\exists\!\!\!\sim$-rule; in the other, an r-link is created along with an extra one which is then set as more preferred to it (in $\rho^r_{\mathcal{A}}$).

The $\sqsubseteq\!\!\!\sim$-rule handles the presence of DCIs in the knowledge base, which have a global behaviour just as the GCIs in $\mathcal{T}$. Given an individual name a, it abides by a DCI $C \sqsubseteq\!\!\!\sim_r D$ if at least one of the following three possibilities holds: (*i*) a is not in C; or (*ii*) a falls under C but there is another instance of C that is more preferred than a, or (*iii*) a is in D. This is captured by the or-like branch in the rule. Moreover, we need to check whether the node is not blocked in order to prevent the creation of an infinitely descending chain of increasingly more preferred objects. (This is needed to ensure termination of the algorithm and also that the preference relation on pairs of objects created when unraveling an open tableau is well-founded.)

The min-rule ensures that every individual that is more preferred than a typical instance of C is marked as an instance of $\neg C$.

Finally, the $\ll$-rule takes care of completing $\rho^r_{\mathcal{A}}$ based on the information in $\sigma^r_{\mathcal{A}}$ so that the ordering on objects induced by that on pairs that $\rho^r_{\mathcal{A}}$ gives rise to coincides with the ordering on objects given by the strict version of $\sigma^r_{\mathcal{A}}$. (See also Definition 3.) This is needed because at the end of the tableau execution, $\sigma^r_{\mathcal{A}}$ is discarded and only $\rho^r_{\mathcal{A}}$ is used to define an ordering on objects against which to check satisfiability of DCIs.

Definition 10 (Complete and clash-free ABox). *Let $\mathcal{A}$ be an ABox. We say $\mathcal{A}$ contains a **clash** if there is some $a \in \mathsf{I}$ and $C \in \mathcal{L}_{d\mathcal{ALC}}$ such that $\{a : C, a : \neg C\} \subseteq \mathcal{A}$. We say $\mathcal{A}$ is **clash-free** if it does not contain a clash. $\mathcal{A}$ is **complete** if it contains a clash or if none of the expansion rules in Figure 4 is applicable to $\mathcal{A}$.*

Let $\mathsf{ndexp}(\cdot)$ denote a function taking as input a clash-free ABox $\mathcal{A}$, a nondeterministic rule $\mathbf{R}$ from Figure 4, and an assertion $\alpha \in \mathcal{A}$ such that $\mathbf{R}$ is applicable to α in $\mathcal{A}$. In our case, the nondeterministic rules are the $\sqcup$-, $\exists$- and $\sqsubseteq\!\!\!\sim$-rules. The function returns a set $\mathsf{ndexp}(\mathcal{A}, \mathbf{R}, \alpha)$ containing each of the possible ABoxes resulting from the application of $\mathbf{R}$ to α in $\mathcal{A}$.

The tableau-based procedure for checking consistency of a $d\mathcal{ALC}$ knowledge base $\mathcal{KB} = \mathcal{T} \cup \mathcal{D} \cup \mathcal{A}$ is given in Algorithm 1 below. It uses Function Expand to apply the rules in Figure 4 to $\mathcal{A}$ w.r.t. $\mathcal{T}$ and $\mathcal{D}$. Given an ABox $\mathcal{A}$, with $\rho_{\mathcal{A}}$, $\sigma_{\mathcal{A}}$ and $\tau_{\mathcal{A}}$ we denote, respectively, the sequences $\langle \rho^{r_1}_{\mathcal{A}}, \ldots, \rho^{r_{\#\mathsf{R}}}_{\mathcal{A}} \rangle$, $\langle \sigma^{r_1}_{\mathcal{A}}, \ldots, \sigma^{r_{\#\mathsf{R}}}_{\mathcal{A}} \rangle$ and $\langle \tau^{r_1}_{\mathcal{A}}, \ldots, \tau^{r_{\#\mathsf{R}}}_{\mathcal{A}} \rangle$.

Algorithm 1: Consistent($\mathcal{KB}$)

Input: A $d\mathcal{ALC}$ knowledge base $\mathcal{KB} = \mathcal{T} \cup \mathcal{D} \cup \mathcal{A}$

if Expand($\mathcal{KB}$) $\neq \emptyset$ **then**
    return "Consistent"
else
    return "Inconsistent"

Function Expand($\mathcal{KB}$)

Input: A $d\mathcal{ALC}$ knowledge base $\mathcal{KB} = \mathcal{T} \cup \mathcal{D} \cup \mathcal{A}$

if $\mathcal{A}$ *is not complete* **then**
    Select a rule $\mathbf{R}$ that is applicable to $\mathcal{A}$;
    if $\mathbf{R}$ *is a nondeterministic rule* **then**
        Select an assertion $\alpha \in \mathcal{A}$ to which $\mathbf{R}$ is applicable;
        if *there is* $\mathcal{A}' \in \mathsf{ndexp}(\mathcal{A}, \mathbf{R}, \alpha)$ *with* Expand($\mathcal{T} \cup \mathcal{D} \cup \mathcal{A}'$) $\neq \emptyset$ **then**
            return Expand($\mathcal{T} \cup \mathcal{D} \cup \mathcal{A}'$)
        else
            return $\emptyset$
    else
        Apply $\mathbf{R}$ to $\mathcal{A}$
if $\mathcal{A}$ *contains a clash* **then**
    return $\emptyset$
else
    return $\langle \mathcal{A}, \rho_{\mathcal{A}}, \sigma_{\mathcal{A}}, \tau_{\mathcal{A}} \rangle$

Lemma 2 (Termination). *For every knowledge base $\mathcal{KB}$,* Consistent($\mathcal{KB}$) *terminates.*

The proof of Lemma 2 is similar to that showing termination of the classical $\mathcal{ALC}$ tableau for checking consistency of general knowledge bases [4, Lemma 4.10].

Theorem 1. *Algorithm 1 is sound and complete w.r.t. preferential consistency of $d\mathcal{ALC}$ knowledge bases.*

Corollary 1. *Our tableau-based algorithm is a decision procedure for satisfiability of $d\mathcal{ALC}$ knowledge bases.*

5 Concluding Remarks

The tableau procedure presented here can be implemented as a proof procedure for checking consistency of contextual defeasible knowledge bases. It can also be used to perform preferential (and modular) entailment checking, and hence also used as part of an algorithm to determine rational closure. In its current form the complexity of the naïve procedure is doubly-exponential, with an optimal proof procedure currently under investigation.

References

1. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. (eds.): The Description Logic Handbook: Theory, Implementation and Applications. Cambridge University Press, 2 edn. (2007)
2. Baader, F., Hollunder, B.: How to prefer more specific defaults in terminological default logic. In: Bajcsy, R. (ed.) Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI). pp. 669–675. Morgan Kaufmann Publishers (1993)
3. Baader, F., Hollunder, B.: Embedding defaults into terminological knowledge representation formalisms. Journal of Automated Reasoning **14**(1), 149–180 (1995)
4. Baader, F., Horrocks, I., Lutz, C., Sattler, U.: An Introduction to Description Logic. Cambridge University Press (2017)
5. Bonatti, P., Faella, M., Petrova, I., Sauro, L.: A new semantics for overriding in description logics. Artificial Intelligence **222**, 1–48 (2015)
6. Bonatti, P., Faella, M., Sauro, L.: Defeasible inclusions in low-complexity DLs. Journal of Artificial Intelligence Research **42**, 719–764 (2011)
7. Bonatti, P., Lutz, C., Wolter, F.: The complexity of circumscription in description logic. Journal of Artificial Intelligence Research **35**, 717–773 (2009)
8. Booth, R., Casini, G., Meyer, T., Varzinczak, I.: On the entailment problem for a logic of typicality. In: Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI). pp. 2805–2811 (2015)
9. Booth, R., Meyer, T., Varzinczak, I.: PTL: A propositional typicality logic. In: Fariñas del Cerro, L., Herzig, A., Mengin, J. (eds.) Proceedings of the 13th European Conference on Logics in Artificial Intelligence (JELIA). pp. 107–119. No. 7519 in LNCS, Springer (2012)
10. Booth, R., Meyer, T., Varzinczak, I.: A propositional typicality logic for extending rational consequence. In: Fermé, E., Gabbay, D., Simari, G. (eds.) Trends in Belief Revision and Argumentation Dynamics, Studies in Logic – Logic and Cognitive Systems, vol. 48, pp. 123–154. King's College Publications (2013)
11. Boutilier, C.: Conditional logics of normality: A modal approach. Artificial Intelligence **68**(1), 87–154 (1994)
12. Britz, K., Casini, G., Meyer, T., Varzinczak, I.: Preferential role restrictions. In: Proceedings of the 26th International Workshop on Description Logics. pp. 93–106 (2013)
13. Britz, K., Heidema, J., Meyer, T.: Semantic preferential subsumption. In: Lang, J., Brewka, G. (eds.) Proceedings of the 11th International Conference on Principles of Knowledge Representation and Reasoning (KR). pp. 476–484. AAAI Press/MIT Press (2008)
14. Britz, K., Heidema, J., Meyer, T.: Modelling object typicality in description logics. In: Nicholson, A., Li, X. (eds.) Proceedings of the 22nd Australasian Joint Conference on Artificial Intelligence. pp. 506–516. No. 5866 in LNAI, Springer (2009)
15. Britz, K., Meyer, T., Varzinczak, I.: Semantic foundation for preferential description logics. In: Wang, D., Reynolds, M. (eds.) Proceedings of the 24th Australasian Joint Conference on Artificial Intelligence. pp. 491–500. No. 7106 in LNAI, Springer (2011)
16. Britz, K., Varzinczak, I.: Defeasible modalities. In: Proceedings of the 14th Conference on Theoretical Aspects of Rationality and Knowledge (TARK). pp. 49–60 (2013)
17. Britz, K., Varzinczak, I.: Introducing role defeasibility in description logics. In: Michael, L., Kakas, A. (eds.) Proceedings of the 15th European Conference on Logics in Artificial Intelligence (JELIA). pp. 174–189. No. 10021 in LNCS, Springer (2016)
18. Britz, K., Varzinczak, I.: Context-based defeasible subsumption for $d\mathcal{SROIQ}$. In: Proceedings of the 13th International Symposium on Logical Formalizations of Commonsense Reasoning (2017)

19. Britz, K., Varzinczak, I.: From KLM-style conditionals to defeasible modalities, and back. Journal of Applied Non-Classical Logics (JANCL) **28**(1), 92–121 (2018)
20. Britz, K., Varzinczak, I.: Preferential accessibility and preferred worlds. Journal of Logic, Language and Information (JoLLI) **27**(2), 133–155 (2018)
21. Britz, K., Varzinczak, I.: Rationality and context in defeasible subsumption. In: Ferrarotti, F., Woltran, S. (eds.) Proceedings of the 10th International Symposium on Foundations of Information and Knowledge Systems (FoIKS). pp. 114–132. No. 10833 in LNCS, Springer (2018)
22. Casini, G., Meyer, T., Moodley, K., Sattler, U., Varzinczak, I.: Introducing defeasibility into OWL ontologies. In: Arenas, M., Corcho, O., Simperl, E., Strohmaier, M., d'Aquin, M., Srinivas, K., Groth, P., Dumontier, M., Heflin, J., Thirunarayan, K., Staab, S. (eds.) Proceedings of the 14th International Semantic Web Conference (ISWC). pp. 409–426. No. 9367 in LNCS, Springer (2015)
23. Casini, G., Straccia, U.: Rational closure for defeasible description logics. In: Janhunen, T., Niemelä, I. (eds.) Proceedings of the 12th European Conference on Logics in Artificial Intelligence (JELIA). pp. 77–90. No. 6341 in LNCS, Springer-Verlag (2010)
24. Casini, G., Straccia, U.: Defeasible inheritance-based description logics. Journal of Artificial Intelligence Research (JAIR) **48**, 415–473 (2013)
25. Donini, F., Nardi, D., Rosati, R.: Description logics of minimal knowledge and negation as failure. ACM Transactions on Computational Logic **3**(2), 177–225 (2002)
26. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.: Preferential description logics. In: Dershowitz, N., Voronkov, A. (eds.) Logic for Programming, Artificial Intelligence, and Reasoning (LPAR). pp. 257–272. No. 4790 in LNAI, Springer (2007)
27. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.: Reasoning about typicality in preferential description logics. In: Hölldobler, S., Lutz, C., Wansing, H. (eds.) Proceedings of the 11th European Conference on Logics in Artificial Intelligence (JELIA). pp. 192–205. No. 5293 in LNAI, Springer (2008)
28. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.: $\mathcal{ALC} + T$: a preferential extension of description logics. Fundamenta Informaticae **96**(3), 341–372 (2009)
29. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.: A non-monotonic description logic for reasoning about typicality. Artificial Intelligence **195**, 165–202 (2013)
30. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.: Semantic characterization of rational closure: From propositional logic to description logics. Artificial Intelligence **226**, 1–33 (2015)
31. Governatori, G.: Defeasible description logics. In: Antoniou, G., Boley, H. (eds.) Rules and Rule Markup Languages for the Semantic Web. pp. 98–112. No. 3323 in LNCS, Springer (2004)
32. Grosof, B., Horrocks, I., Volz, R., Decker, S.: Description logic programs: Combining logic programs with description logic. In: Proceedings of the 12th International Conference on World Wide Web (WWW). pp. 48–57. ACM (2003)
33. Heymans, S., Vermeir, D.: A defeasible ontology language. In: Meersman, R., Tari, Z. (eds.) CoopIS/DOA/ODBASE. pp. 1033–1046. No. 2519 in LNCS, Springer (2002)
34. Kraus, S., Lehmann, D., Magidor, M.: Nonmonotonic reasoning, preferential models and cumulative logics. Artificial Intelligence **44**, 167–207 (1990)
35. Lehmann, D., Magidor, M.: What does a conditional knowledge base entail? Artificial Intelligence **55**, 1–60 (1992)
36. Padgham, L., Zhang, T.: A terminological logic with defaults: A definition and an application. In: Bajcsy, R. (ed.) Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI). pp. 662–668. Morgan Kaufmann Publishers (1993)
37. Qi, G., Pan, J., Ji, Q.: Extending description logics with uncertainty reasoning in possibilistic logic. In: Mellouli, K. (ed.) Proceedings of the 9th European Conference on Symbolic and

Quantitative Approaches to Reasoning with Uncertainty. pp. 828–839. No. 4724 in LNAI, Springer (2007)
38. Quantz, J., Royer, V.: A preference semantics for defaults in terminological logics. In: Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning (KR). pp. 294–305 (1992)
39. Quantz, J., Ryan, M.: Preferential default description logics. Tech. rep., TU Berlin (1993), www.tu-berlin.de/fileadmin/fg53/KIT-Reports/r110.pdf
40. Sengupta, K., Alfa Krisnadhi, A., Hitzler, P.: Local closed world semantics: Grounded circumscription for OWL. In: Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N., Blomqvist, E. (eds.) Proceedings of the 10th International Semantic Web Conference (ISWC). pp. 617–632. No. 7031 in LNCS, Springer (2011)
41. Shoham, Y.: Reasoning about Change: Time and Causation from the Standpoint of Artificial Intelligence. MIT Press (1988)
42. Straccia, U.: Default inheritance reasoning in hybrid KL-ONE-style logics. In: Bajcsy, R. (ed.) Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI). pp. 676–681. Morgan Kaufmann Publishers (1993)
43. Varzinczak, I.: A note on a description logic of concept and role typicality for defeasible reasoning over ontologies. Logica Universalis **12**(3-4), 297–325 (2018)