Nonmonotonic Reasoning in Description Logic by Tableaux Algorithm with Blocking

Jaromír Malenko and Petr Štěpánek

Charles University, Malostranske namesti 25, 11800 Prague, Czech Republic,
Jaromir.Malenko@mff.cuni.cz, Petr.Stepanek@mff.cuni.cz

Abstract. To support nonmonotonic reasoning we introduce the description logic of minimal knowledge and negation as failure (MKNF-DL) as an extension of description logic with modal operators **K** and **A**. We discuss the problems with representation of a model for an MKNF-DL theory. For satisfiability checking of MKNF-DL theories, we introduce a tableaux algorithm with blocking, where blocking works with the modal part of an MKNF-DL theory. This blocking technique allows for reasoning about a larger class of MKNF-DL theories than previous approaches.

Recently, Description Logics (DL) are used to represent and reason about knowledge bases (KBs). In practical applications, the monotonic property of standard logics, which includes DLs, may be undesirable.

Hence, we introduce $\mathcal{ALCK}_{\mathcal{NF}}$, the DL of minimal knowledge and negation as failure (MKNF-DL) as an extension of description logic (DL) with modal operators **K** and **A**. Advanced reasoning applications, including epistemic queries, integrity constraints and default rules can be represented by $\mathcal{ALCK}_{\mathcal{NF}}$ [4].

Next, we introduce a reasoning technique for $\mathcal{ALCK}_{\mathcal{NF}}$. To this end the representation of $\mathcal{ALCK}_{\mathcal{NF}}$ models is crucial. The models of $\mathcal{ALCK}_{\mathcal{NF}}$ are not first-order representable. Hence, we define *subjectively quantified* KBs which are representable by $\mathcal{ALC}$ theory. However, this $\mathcal{ALC}$ theory may be infinite. Previous approaches [1, 3] defined *simple* KBs, a subset of subjectively quantified KBs, which are representable by finite $\mathcal{ALC}$ theory. The intention of our research was an effective reasoning method for subjectively quantified KB. We achieved this by introducing a tableaux algorithm with blocking; however, for the algorithm to be complete, a further restriction to *minimality-proper* KBs is neccessary. Still, minimality-proper KBs include simple KBs.

1 Basic Formalism

We start with usual formalism of MKNF-DL, which roughly follow Donini [1].

1.1 Syntax

Definition 1 ($\mathcal{ALCK}_{\mathcal{NF}}$ Syntax) *The $\mathcal{ALCK}_{\mathcal{NF}}$ syntax is defined as follows:*

$$C ::= \ \top \mid \bot \mid C_a \mid \neg C \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \exists R.C \mid \forall R.C \mid \mathbf{K}C \mid \mathbf{A}C$$
$$R ::= \ R_a \mid \mathbf{K}R \mid \mathbf{A}R$$

where C_a is an atomic concept, C, C_1, C_2 are concept expressions, R_a is an atomic role, R is a role, and **K** *and* **A** *are modal operators.*

Definition 2 *An $\mathcal{ALCK_{NF}}$ concept C is subjective if each $\mathcal{ALC}$ atomic subconcept of C is also a subconcept of a modal subconcept of C. An $\mathcal{ALCK_{NF}}$ concept C is objective if C does not contain a modal role or a modal subconcept. An $\mathcal{ALCK_{NF}}$ concept C is mixed if C is neither subjective nor objective.*

Objective concepts are the $\mathcal{ALC}$ concepts. For example, concept $\exists\mathbf{K}R.C$ is neither subjective nor objective.

The $\mathcal{ALCK_{NF}}$ knowledge base (KB) is an extension of $\mathcal{ALC}$ KB to which we add the modal formulae.

Definition 3 ($\mathcal{ALCK_{NF}}$ Knowledge Base) *The $\mathcal{ALCK_{NF}}$ knowledge base Σ is a triple $\langle \mathcal{T}, \Gamma, \mathcal{A} \rangle$, where TBox $\mathcal{T}$ is a set of objective axioms, MBox Γ is a set of non-objective axioms, and ABox $\mathcal{A}$ is a set of (both objective and non-objective) assertions.*

In the remainder we consider the following simplification of notation. We assume an $\mathcal{ALCK_{NF}}$ KB $\Sigma = \langle \mathcal{T}, \Gamma, \mathcal{A} \rangle$. **M** denotes a modal operator, i.e., $\mathbf{M} \in \{\mathbf{K}, \mathbf{A}\}$, **N** denotes a possibly negated modal operator, i.e., $\mathbf{N} \in \{\mathbf{K}, \mathbf{A}, \neg\mathbf{K}, \neg\mathbf{A}\}$. R_a denotes an atomic $\mathcal{ALC}$ role, C_a, D_a denote atomic $\mathcal{ALC}$ concepts, A, B, C, D (possibly with indices or primes) denote arbitrary $\mathcal{ALCK_{NF}}$ concepts.

1.2 Semantics

Remind, that an $\mathcal{ALC}$ interpretation $\mathcal{I} = (\Delta, \cdot^{\mathcal{I}})$ consists of *domain* Δ and *interpretation function* $\cdot^{\mathcal{I}}$ which maps each atomic concept C_a to a subset of domain: $C_a^{\mathcal{I}} \subseteq \Delta$, each atomic role R_a to a subset of cartesian product of domain: $R_a^{\mathcal{I}} \subseteq \Delta \times \Delta$, and each individual name x to an element of domain: $x^{\mathcal{I}} \in \Delta$.

The semantics of $\mathcal{ALCK_{NF}}$ considers sets of $\mathcal{ALC}$ interpretations with the following properties:

(i) The domain Δ is a countable (possibly infinite) set.
(ii) All $\mathcal{ALC}$ interpretations are defined over the same domain Δ.
(iii) Each individual name in every interpretation maps to the same domain element.

Definition 4 ($\mathcal{ALCK_{NF}}$ Semantics) *An $\mathcal{ALCK_{NF}}$ interpretation is a triple $(\mathcal{I}, \mathcal{M}, \mathcal{N})$, where $\mathcal{I}$ is an $\mathcal{ALC}$ interpretation $(\Delta, \cdot^{\mathcal{I}})$, and $\mathcal{M}$ and $\mathcal{N}$ are sets of $\mathcal{ALC}$ interpretations.*

Atomic concepts C_a, roles R_a, and individuals a are interpreted in $\mathcal{I}$ as usual in $\mathcal{ALC}$:

$$(C_a)^{\mathcal{I},\mathcal{M},\mathcal{N}} = C_a^{\mathcal{I}}$$
$$(R_a)^{\mathcal{I},\mathcal{M},\mathcal{N}} = R_a^{\mathcal{I}}$$
$$(a)^{\mathcal{I},\mathcal{M},\mathcal{N}} = a^{\mathcal{I}}.$$

The $\mathcal{ALCK_{NF}}$ interpretation $(\mathcal{I}, \mathcal{M}, \mathcal{N})$ is extended to non-atomic and modal concepts and modal roles as follows:

$$
\begin{aligned}
(\top)^{\mathcal{I},\mathcal{M},\mathcal{N}} &= \Delta \\
(\bot)^{\mathcal{I},\mathcal{M},\mathcal{N}} &= \emptyset \\
(\neg C)^{\mathcal{I},\mathcal{M},\mathcal{N}} &= \Delta \setminus (C)^{\mathcal{I},\mathcal{M},\mathcal{N}} \\
(C_1 \sqcap C_2)^{\mathcal{I},\mathcal{M},\mathcal{N}} &= (C_1)^{\mathcal{I},\mathcal{M},\mathcal{N}} \cap (C_2)^{\mathcal{I},\mathcal{M},\mathcal{N}} \\
(C_1 \sqcup C_2)^{\mathcal{I},\mathcal{M},\mathcal{N}} &= (C_1)^{\mathcal{I},\mathcal{M},\mathcal{N}} \cup (C_2)^{\mathcal{I},\mathcal{M},\mathcal{N}} \\
(\exists R.C)^{\mathcal{I},\mathcal{M},\mathcal{N}} &= \{x \in \Delta \mid \exists y \in \Delta : (x,y) \in (R)^{\mathcal{I},\mathcal{M},\mathcal{N}} \wedge y \in (C)^{\mathcal{I},\mathcal{M},\mathcal{N}}\} \\
(\forall R.C)^{\mathcal{I},\mathcal{M},\mathcal{N}} &= \{x \in \Delta \mid \forall y \in \Delta : (x,y) \in (R)^{\mathcal{I},\mathcal{M},\mathcal{N}} \Rightarrow y \in (C)^{\mathcal{I},\mathcal{M},\mathcal{N}}\} \\
(\mathbf{K}C)^{\mathcal{I},\mathcal{M},\mathcal{N}} &= \bigcap\nolimits_{\mathcal{J} \in \mathcal{M}} (C)^{\mathcal{J},\mathcal{M},\mathcal{N}} \\
(\mathbf{A}C)^{\mathcal{I},\mathcal{M},\mathcal{N}} &= \bigcap\nolimits_{\mathcal{J} \in \mathcal{N}} (C)^{\mathcal{J},\mathcal{M},\mathcal{N}} \\
(\mathbf{K}R_a)^{\mathcal{I},\mathcal{M},\mathcal{N}} &= \bigcap\nolimits_{\mathcal{J} \in \mathcal{M}} (R_a)^{\mathcal{J},\mathcal{M},\mathcal{N}} \\
(\mathbf{A}R_a)^{\mathcal{I},\mathcal{M},\mathcal{N}} &= \bigcap\nolimits_{\mathcal{J} \in \mathcal{N}} (R_a)^{\mathcal{J},\mathcal{M},\mathcal{N}}
\end{aligned}
$$

Definition 5 (Satisfiability in $(\mathcal{I}, \mathcal{M}, \mathcal{N})$) *A concept inclusion $C \sqsubseteq D$ is satisfied in $(\mathcal{I}, \mathcal{M}, \mathcal{N})$, denoted as $(\mathcal{I}, \mathcal{M}, \mathcal{N}) \models C \sqsubseteq D$, iff $C^{\mathcal{I},\mathcal{M},\mathcal{N}} \subseteq D^{\mathcal{I},\mathcal{M},\mathcal{N}}$. A concept assertion $C(a)$ is satisfied in $(\mathcal{I}, \mathcal{M}, \mathcal{N})$, denoted as $(\mathcal{I}, \mathcal{M}, \mathcal{N}) \models C(a)$, iff $a \in C^{\mathcal{I},\mathcal{M},\mathcal{N}}$. A role assertion $R(a,b)$ is satisfied in $(\mathcal{I}, \mathcal{M}, \mathcal{N})$, denoted as $(\mathcal{I}, \mathcal{M}, \mathcal{N}) \models R(a,b)$, iff $(a,b) \in R^{\mathcal{I},\mathcal{M},\mathcal{N}}$.*

Definition 6 (Satisfiability in $(\mathcal{M}, \mathcal{N})$) *A concept inclusion $C \sqsubseteq D$ is satisfied in $(\mathcal{M}, \mathcal{N})$, denoted as $(\mathcal{M}, \mathcal{N}) \models C \sqsubseteq D$, iff $C^{\mathcal{I},\mathcal{M},\mathcal{N}} \subseteq D^{\mathcal{I},\mathcal{M},\mathcal{N}}$ for each $\mathcal{I} \in \mathcal{M}$. A concept assertion $C(a)$ is satisfied in $(\mathcal{M}, \mathcal{N})$, denoted as $(\mathcal{M}, \mathcal{N}) \models C(a)$, iff $a \in C^{\mathcal{I},\mathcal{M},\mathcal{N}}$ for each $\mathcal{I} \in \mathcal{M}$. A role assertion $R(a,b)$ is satisfied in $(\mathcal{M}, \mathcal{N})$, denoted as $(\mathcal{M}, \mathcal{N}) \models R(a,b)$, iff $(a,b) \in R^{\mathcal{I},\mathcal{M},\mathcal{N}}$ for each $\mathcal{I} \in \mathcal{M}$.*

A TBox $\mathcal{T}$ is satisfied in $(\mathcal{M}, \mathcal{N})$, denoted as $(\mathcal{M}, \mathcal{N}) \models \mathcal{T}$, iff all axioms in $\mathcal{T}$ are satisfied in $(\mathcal{M}, \mathcal{N})$. An MBox Γ is satisfied in $(\mathcal{M}, \mathcal{N})$, denoted as $(\mathcal{M}, \mathcal{N}) \models \Gamma$, iff all axioms in Γ are satisfied in $(\mathcal{M}, \mathcal{N})$. An ABox $\mathcal{A}$ is satisfied in $(\mathcal{M}, \mathcal{N})$, denoted as $(\mathcal{M}, \mathcal{N}) \models \mathcal{A}$, iff all assertions in $\mathcal{A}$ are satisfied in $(\mathcal{M}, \mathcal{N})$. A KB Σ is satisfied in $(\mathcal{M}, \mathcal{N})$, denoted as $(\mathcal{M}, \mathcal{N}) \models \Sigma$, iff all $\mathcal{T}$, Γ and $\mathcal{A}$ are satisfied in $(\mathcal{M}, \mathcal{N})$.

Up to now, the definition of modal operators $\mathbf{K}$ and $\mathbf{A}$ were the same. Their meaning is distinguished by imposing the maximality condition on $\mathbf{K}$.

Definition 7 ($\mathcal{ALCK_{NF}}$ Model) *A set of $\mathcal{ALC}$ interpretations $\mathcal{M}$ is a model for an $\mathcal{ALCK_{NF}}$ KB Σ iff the following two conditions hold:*

(i) the structure $(\mathcal{M}, \mathcal{M})$ satisfies Σ; and
(ii) for each set of $\mathcal{ALC}$ interpretations $\mathcal{M}'$, if $\mathcal{M}' \supset \mathcal{M}$ then the structure $(\mathcal{M}', \mathcal{M})$ does not satisfy Σ.

Definition 8 (Entailment) *A concept inclusion $C \sqsubseteq D$ is a consequence of KB Σ, denoted as $\Sigma \models C \sqsubseteq D$, iff $(\mathcal{M}, \mathcal{M}) \models C \sqsubseteq D$ for each model $\mathcal{M}$ of Σ. A concept assertion $C(a)$ is a consequence of KB Σ, denoted as $\Sigma \models$*

$C(a)$, iff $(\mathcal{M}, \mathcal{M}) \models C(a)$ for each model $\mathcal{M}$ of Σ. A role assertion $R(a, b)$ is a consequence of KB Σ, denoted as $\Sigma \models R(a, b)$, iff $(\mathcal{M}, \mathcal{M}) \models R(a, b)$ for each model $\mathcal{M}$ of Σ.

Definition 9 (Satisfiability) *A KB Σ is satisfiable if there is a model for Σ. Two KBs Σ and Σ' are equivalent, denoted as $\Sigma \equiv \Sigma'$, iff for every set of $\mathcal{ALC}$ interpretations $\mathcal{M}$, $\mathcal{M}$ is model for Σ iff $\mathcal{M}$ is model for Σ'. Two concepts C and C' are equivalent in KB Σ, denoted as $\Sigma \models C \equiv C'$, iff for every set of $\mathcal{ALC}$ interpretations $\mathcal{M}$ of Σ, $\mathcal{M} \models C \sqsubseteq C'$ and $\mathcal{M} \models C' \sqsubseteq C$.*

2 Model Representation

Our intention is to introduce a reasoning technique for $\mathcal{ALCK}_{\mathcal{NF}}$. This will be attained in the next section. In this section, we discuss the *representation* of $\mathcal{ALCK}_{\mathcal{NF}}$ models.

In our treatment of representing and reasoning in $\mathcal{ALCK}_{\mathcal{NF}}$ we follow several approaches to reasoning in propositional modal logic [2, 5, 6].

The reasoning problem for $\mathcal{ALCK}_{\mathcal{NF}}$ is reduced to several reasoning problems in the underlying nonmodal logic $\mathcal{ALC}$. Each model for an $\mathcal{ALCK}_{\mathcal{NF}}$ KB Σ is characterized by an $\mathcal{ALC}$ KB. Therefore, the set of $\mathcal{ALC}$ KBs that represents all the models of Σ constitute a nonmodal representation of Σ. Such a representation allows for using classical reasoning techniques for $\mathcal{ALC}$ to solve the reasoning problems for $\mathcal{ALCK}_{\mathcal{NF}}$.

Recall that an $\mathcal{ALCK}_{\mathcal{NF}}$ model $\mathcal{M}$ is a set of $\mathcal{ALC}$ interpretations. The model $\mathcal{M}$ is *first-order representable* if there exists a first-order theory (resp. $\mathcal{ALC}$ KB) $\Sigma_{\mathcal{M}}$ such that

$$\mathcal{M} = \{\mathcal{I} : \mathcal{I} \models \Sigma_{\mathcal{M}}\}.$$
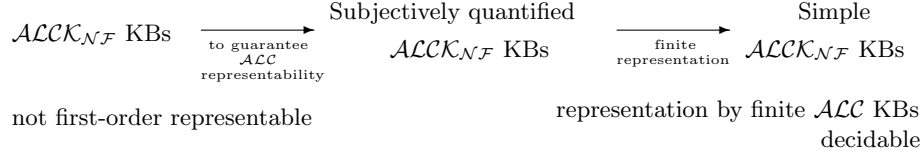
Therefore the set of interpretations belonging to $\mathcal{M}$ can be represented by a first-order theory $\Sigma_{\mathcal{M}}$. Note that the theory $\Sigma_{\mathcal{M}}$ may be either finite or infinite.

Following the motivation of this approach, we consider only the case when the $\mathcal{ALCK}_{\mathcal{NF}}$ model $\mathcal{M}$ is $\mathcal{ALC}$ *representable*, i.e. there exists an $\mathcal{ALC}$ KB $\Sigma_{\mathcal{M}}$ such that $\mathcal{M} = \{\mathcal{I} : \mathcal{I} \models \Sigma_{\mathcal{M}}\}$. Moreover, we consider only the case when the corresponding $\Sigma_{\mathcal{M}}$ is finite. Since $\mathcal{ALC}$ is a fragment of first-order logic, if $\mathcal{M}$ is $\mathcal{ALC}$ representable then $\mathcal{M}$ is also first-order representable.

To guarantee $\mathcal{ALC}$ representability of $\mathcal{ALCK}_{\mathcal{NF}}$ KB models, the KB is restricted to a *subjectively quantified KB*. In previous publications [1, 3], a subjectively quantified KB is further restricted to *simple KB* whose models admit a representation in terms of *finite $\mathcal{ALC}$ KB* and guarantee decidability of reasoning (specifically termination of the tableaux algorithm). This progress is depicted in Figure 1.

The approach of subjectively quantified KBs and simple KBs was first introduced and extensively analysed in Donini et al. [1]. A further research was done by Ke et al. [3]. We now compare both papers and explain our approach.

First, Donini et al. [1] claim that the models of $\mathcal{ALCK}_{\mathcal{NF}}$ KBs cannot be characterized by first-order theories.

$\mathcal{ALCK}_{\mathcal{NF}}$ KBs $\xrightarrow[\substack{\text{to guarantee}\\ \mathcal{ALC}\\ \text{representability}}]{}$ Subjectively quantified $\mathcal{ALCK}_{\mathcal{NF}}$ KBs $\xrightarrow[\substack{\text{finite}\\ \text{representation}}]{}$ Simple $\mathcal{ALCK}_{\mathcal{NF}}$ KBs

not first-order representable

representation by finite $\mathcal{ALC}$ KBs
decidable

Fig. 1. The structure of work on $\mathcal{ALCK}_{\mathcal{NF}}$ model representation. In previous publications [1, 3], to obtain a reasoning algorithm, the language of KBs was syntactically restricted to subjectively quantified and then to simple KB. In this thesis, we provide a reasoning algorithm that works with subjectively quantified KBs. Since our approach does not require the simple restriction, it allows reasoning over greater set of KBs.

Theorem 10 *The models of $\mathcal{ALCK}_{\mathcal{NF}}$ are not first-order representable.*

Briefly, the proof considers a carefully constructed $\mathcal{ALCK}_{\mathcal{NF}}$ KB Σ, shows a model $\mathcal{M}$ for Σ, and shows that $\mathcal{M}$ cannot be characterized in terms of first-order theory.

Second, they identify a subset of $\mathcal{ALCK}_{\mathcal{NF}}$ KBs whose models are first-order representable by means of (either finite or infinite) first-order theories. This is achieved by the notion of subjectively quantified KBs. We consider two definitions from previous papers [1, 3].

Definition 11 (Subjectively Quantified KB) *A subjectively quantified $\mathcal{ALCK}_{\mathcal{NF}}$ KB is an $\mathcal{ALCK}_{\mathcal{NF}}$ KB Σ such that each concept C of the form $\exists R.D$ or $\forall R.D$ satisfies one of the following conditions:*

(i) R is an $\mathcal{ALC}$ role and D is an $\mathcal{ALC}$ concept;
(ii) R is of the form $\mathbf{M}R_a$ and D is subjective.

To summarize, the Definition 11 by [3] is more general than similar definition by [1], and we use it for the rest of this paper. However the difference between the definitions has no implications for the tableaux algorithm, since the algorithm considers the flatten normal forms of formulae (created in preprocessing stage).

Theorem 12 *A subjectively quantified $\mathcal{ALCK}_{\mathcal{NF}}$ KB is representable by an (either finite or infinite) $\mathcal{ALC}$ theory.*

Briefly, the proof shows that the models of a subjectively quantified $\mathcal{ALCK}_{\mathcal{NF}}$ KB can be characterized in terms of an $\mathcal{ALC}$ KB. The characterization relies on the notion of *modal atom*: there is a one-to-one correspondence between certain sets of modal atoms and the models of subjectively quantified $\mathcal{ALCK}_{\mathcal{NF}}$ KB.

2.1 Previous Work

In previous publications [1, 3], authors further restrict $\mathcal{ALCK}_{\mathcal{NF}}$ KBs to obtain a finite characterization of models (that are also first-order representable). This is achieved by the notion of simple KBs.

In the following, we say that an $\mathcal{ALCK}_{\mathcal{NF}}$ concept is *simple* if C is subjectively quantified and each quantified concept subexpression of the form $\exists\mathbf{A}R.\mathbf{N}D$, $\forall\mathbf{A}R.\mathbf{N}D$, where $\mathbf{N} \in \{\mathbf{K}, \neg\mathbf{K}, \mathbf{A}, \neg\mathbf{A}\}$, occurring in C is such that in D there are no occurrences of role expressions of the form $\mathbf{K}R$.

Definition 13 (Simple KB) *A simple $\mathcal{ALCK}_{\mathcal{NF}}$ KB $\Sigma = \langle \mathcal{T}, \Gamma, \mathcal{A} \rangle$ is an $\mathcal{ALCK}_{\mathcal{NF}}$ KB that satisfies the following conditions:*

(i) Γ is a set of $\mathcal{ALCK}_{\mathcal{NF}}$ simple inclusions, i.e. inclusions assertions of the form $\mathbf{K}C \sqsubseteq D$, where C is an $\mathcal{ALC}$-concept such that $\mathcal{T} \not\models \top \sqsubseteq C$, and D is a subjectively quantified concept expression in which there are no occurrences of the operator $\mathbf{K}$ within the scope of quantifiers; and

(ii) $\mathcal{A}$ is a set of instance assertions such that all concept subexpressions occurring in $\mathcal{A}$ are simple.

Ke et al. [3] significantly loosens the definition of a simple KB by allowing more concept assertions in $\mathcal{A}$. Both [1,3] conclude with a proof of the following theorem.

Theorem 14 *A simple $\mathcal{ALCK}_{\mathcal{NF}}$ KB is representable by a finite $\mathcal{ALC}$ theory.*

2.2 Our Approach

We present a reasoning algorithm for deciding the satisfiability of a subjectively quantified $\mathcal{ALCK}_{\mathcal{NF}}$ KB. We use a novel tableaux algorithm with a blocking technique to deal with modal part of $\mathcal{ALCK}_{\mathcal{NF}}$ theory.

In standard DLs, the tableaux algorithm tests satisfiability of a concept description. The algorithm starts with a concept description and applies consistency-preserving expansion rules that build a tree representing a model. In the tree, the nodes correspond to individuals and are labelled with sets of DL concept descriptions the individual belongs to.

The models of standard DL KBs can be infinite and therefore the tableaux algorithm uses a blocking technique that identifies infinite branches. This allows to represent the infinite models by finitely-representable models.

The main motivation of our approach is analogous: the the infinite models of $\mathcal{ALCK}_{\mathcal{NF}}$ can be blocked and represented by finitely-representable $\mathcal{ALCK}_{\mathcal{NF}}$ models.

3 Reasoning by Tableaux Algorithm with Blocking

In our approach, we reduce the reasoning problem for $\mathcal{ALCK}_{\mathcal{NF}}$ to several reasoning problems in $\mathcal{ALC}$. Since each model for a subjectively quantified $\mathcal{ALCK}_{\mathcal{NF}}$ KB Σ can be represented by a set of $\mathcal{ALC}$ KBs, this allows us to use classical reasoning techniques for $\mathcal{ALC}$ to solve reasoning problems in $\mathcal{ALCK}_{\mathcal{NF}}$.

In this section we define the tableaux algorithm that tests the satisfiability of a subjectively quantified $\mathcal{ALCK}_{\mathcal{NF}}$ KB $\Sigma = \langle \mathcal{T}, \Gamma, \mathcal{A} \rangle$.

The reader is referred to Malenko et al. [4] for detailed treatment and proofs of theorems in this section.

Since every $\mathcal{ALCK_{NF}}$ KB can be translated to flatten normal form, we assume w.l.o.g. the Σ to be in flatten normal form.

The satisfiability problem for Σ is solved by a tableaux algorithm for Σ. The tableaux algorithm is trying to construct a model for Σ by means of branches which correspond to finite subsets of modal atoms. A branch satisfying certain conditions corresponds to a model for Σ.

The $\mathcal{ALCK_{NF}}$ tableaux algorithm is similar to a standard $\mathcal{ALC}$ tableaux algorithm. However, the are differences: Only modal assertions are decomposed by the $\mathcal{ALCK_{NF}}$ tableaux algorithm, including the GCIs in Γ. An $\mathcal{ALC}$ reasoner is used as an underlying reasoner, which considers $\mathcal{T}$ and $\mathcal{ALC}$ assertions through the so-called objective knowledge of a branch. The $\mathcal{ALC}$ reasoner is called in the conditions of trigger-rule and testing the conditions of models (open branch, pre-preferred branch, minimality condition).

A *branch* $\mathcal{B}$ is a set of membership assertions of the form $C(x)$, where C is an $\mathcal{ALCK_{NF}}$ concept description.

The tableaux starts with an *initial branch* $\mathcal{B}_0 = \{\mathbf{K}C(x) \mid C(x) \in \mathcal{A}\}$. New branches are obtained from the current branch by applying the tableaux expansion rules from Figure 2. We use $\dot{\neg}C$ to denote the negation normal form of the $\neg C$. The set of individuals appearing in $\mathcal{B}$ is denoted by $\mathcal{O}_\mathcal{B}$.

We now briefly describe the $\mathcal{ALCK_{NF}}$ expansion rules and compare them with standard DL tableaux rules [1]:

- The $\sqcap$-rule is analogous to the the corresponding DL tableaux rule.
- The $\sqcup$-rule adds both disjuncts (as they are or possibly negated) to the tableaux, because they are needed in the minimality check. The for parts of the rule consider all the cases that can happen; the part (d) detect and inconsistency and enforces a clash in the underlying DL reasoner. In the corresponding DL tableaux rule, either of the conjunct is added to the tableaux.
- The $\exists$-rule is analogous to the the corresponding DL tableaux rule.
- The $\forall$-rule has two parts. The first part deals with modal roles and is analogous to DL tableaux rule. The second part deals with the relationship between $\mathbf{K}R(x, y)$ and $\mathbf{A}R(x, y)$. In the tableaux algorithm, the "value" of $\mathbf{A}R(x, y)$ is by default "false" until the appearance of $\mathbf{K}R(x, y)$ supporting it to be "true" (this is possible since $\mathbf{A}R(x, y) \in MA_\Delta(\Sigma)$ it is supported by $\forall \mathbf{A}R.C(x) \in MA_\Delta(\Sigma)$).
- The trigger-rule takes Γ into account. The underlying DL reasoner is used to check that $C(a)$ is entailed in $\mathbf{K}$-objective knowledge, which represents "what is known so far" by $\mathcal{T}$ and $\mathcal{B}$.

To employ a blocking technique, the algorithm keeps ordering of individual names according to time of their introduction to tableaux algorithm. Moreover, for each individual y that is introduced into the tableaux by application of the $\exists$-rule to individual x, the algorithm keeps track of the relationship and we define $parent(y) = x$. Then, *predcessor* is transitive closure of *parent*.

$\sqcap$**-rule:**	If $C \sqcap D(x) \in \mathcal{B}$, and $\{C(x), D(x)\} \nsubseteq \mathcal{B}$, then add $C(x), D(x)$ to $\mathcal{B}$.

$\sqcup$**-rule:**	If $C \sqcup D(x) \in \mathcal{B}$ we distinguish the following four cases:
	    (a) if $\{C(x), \dot{\neg}C(x), D(x), \dot{\neg}D(x)\} \cap \mathcal{B} = \emptyset$,
	        then add either $C(x), D(x)$ or $C(x), \dot{\neg}D(x)$ or $\dot{\neg}C(x), D(x)$ to $\mathcal{B}$.
	    (b) if $\{C(x), \dot{\neg}C(x)\} \cap \mathcal{B} = \emptyset$ and $\{D(x), \dot{\neg}D(x)\} \cap \mathcal{B} \neq \emptyset$,
	        then add either $C(x)$ or $\dot{\neg}C(x)$ to $\mathcal{B}$.
	    (c) if $\{C(x), \dot{\neg}C(x)\} \cap \mathcal{B} \neq \emptyset$ and $\{D(x), \dot{\neg}D(x)\} \cap \mathcal{B} = \emptyset$,
	        then add either $D(x)$ or $\dot{\neg}D(x)$ to $\mathcal{B}$.
	    (d) if $\{\dot{\neg}C(x), \dot{\neg}D(x)\} \subseteq \mathcal{B}$,
	        then add $C(x)$ to $\mathcal{B}$.

$\exists$**-rule:**	If $\exists \mathbf{M}R.C(x) \in \mathcal{B}$,
	    and $\{\mathbf{M}R(x, y), C(y)\} \nsubseteq \mathcal{B}$ for any $y \in \mathcal{O}_\mathcal{B}$,
	    and x is not blocked by some $y \in \mathcal{O}_\mathcal{B}$,
	then add $\mathbf{M}R(x, z), C(z)$ to $\mathcal{B}$, for some $z \in \mathcal{O}_\mathcal{B} \cup \{i\}$, where $i \notin \mathcal{O}_\mathcal{B}$.

$\forall$**-rule:**	We distinguish the following two cases:
	    (a) if $\forall \mathbf{M}R.C(x) \in \mathcal{B}$,
	        then for each $\mathbf{M}R(x, y) \in \mathcal{B}$, if $C(y) \notin \mathcal{B}$ then add $C(y)$ to $\mathcal{B}$.
	    (b) if $\forall \mathbf{A}R.C(x) \in \mathcal{B}$,
	        then for each $\mathbf{K}R(x, y) \in \mathcal{B}$, if $\mathbf{A}R(x, y) \notin \mathcal{B}$ then add $\mathbf{A}R(x, y)$ to $\mathcal{B}$.

***trigger*-rule:**	If $\mathbf{K}C_a \sqsubseteq D \in \Gamma, x \in \mathcal{O}_\mathcal{B}, Ob_\mathbf{K}(\mathcal{B}) \models C_a(x)$, and $\{\mathbf{K}C_a(x), D(x)\} \nsubseteq \mathcal{B}$, then add $\mathbf{K}C_a(x), D(x)$ to $\mathcal{B}$.

Fig. 2. Expansion rules for the tableaux algorithm of $\mathcal{ALCK_{NF}}$

Definition 15 (Blocking) *The application of an expansion rule to an individual x is blocked in a branch $\mathcal{B}$ if there is a predecessor y of x and a predecessor z of y such that $\{C \mid C(y) \in \mathcal{B}\} = \{C \mid C(z) \in \mathcal{B}\}$.*

The idea behind blocking is that the blocked individual x can use role successors of y instead of generating new ones. The ordering of individual names according to time of their introduction to tableaux algorithm is considered to avoid cyclic blocking (of x by y and vice versa).

To guarantee termination of this blocking technique, the following strategy for application of expansion rules to the individuals in a branch is used: all rules are applied to the first introduced individual until no more rules apply, then all rules are applied to the second introduced individual until no more rules apply, etc. When a rule is successfully applied to an individual, the rule applications start again from the first individual. When no expansion rules are applicable the tableaux algorithm ends. An effective implementation of this strategy can be achieved by keeping track of tableaux changes and proper backtracking (backjumping).

Definition 16 (Completed Branch) *We say a branch $\mathcal{B}$ is completed if no expansion rules from Figure 2 are applicable to $\mathcal{B}$.*

Definition 17 $\big((P_{\mathcal{B}}, N_{\mathcal{B}})\big)$ *The partition $(P_{\mathcal{B}}, N_{\mathcal{B}})$ associated with a branch $\mathcal{B}$ is defined as follows:*

$$P_{\mathcal{B}} = \{\mathbf{M}C(x) \mid \mathbf{M}C(x) \in \mathcal{B}\} \cup \{\mathbf{M}R(x,y) \mid \mathbf{M}R(x,y) \in \mathcal{B}\}$$
$$N_{\mathcal{B}} = \{\mathbf{M}C(x) \mid \neg \mathbf{M}C(x) \in \mathcal{B}\}$$

Here, $P_{\mathcal{B}}$ is a set of atoms which are true, $N_{\mathcal{B}}$ is a set of atoms which are false. Atoms not in $P_{\mathcal{B}} \cup N_{\mathcal{B}}$ are assumed to be false.

Definition 18 (Objective Knowledge) *Let $\mathcal{B}$ be a branch for Σ. The $\mathcal{ALC}$ KB*

$$Ob_{\mathbf{K}}(\mathcal{B}) = \langle \mathcal{T}, \{C(x) \mid \mathbf{K}C(x) \in P_{\mathcal{B}}\} \cup \{R_a(x,y) \mid \mathbf{K}R_a(x,y) \in P_B\} \rangle$$
$$Ob_{\mathbf{A}}(\mathcal{B}) = \langle \mathcal{T}, \{C(x) \mid \mathbf{A}C(x) \in P_{\mathcal{B}}\} \cup \{R_a(x,y) \mid \mathbf{A}R_a(x,y) \in P_B\} \rangle$$

is called $\mathbf{K}$-objective, resp. $\mathbf{A}$-objective, knowledge for $\mathcal{B}$.

The objective knowledge is an $\mathcal{ALC}$ KB passed to an underlying $\mathcal{ALC}$ reasoner to check a certain set of conditions.

Definition 19 (Open Branch) *A branch $\mathcal{B}$ is open if all of the following conditions hold:*

(i) $Ob_{\mathbf{K}}(\mathcal{B})$ is satisfiable;
(ii) $Ob_{\mathbf{A}}(\mathcal{B})$ is satisfiable;
(iii) $Ob_{\mathbf{K}}(\mathcal{B}) \not\models C(x)$ for each $\mathbf{K}C(x) \in N_{\mathcal{B}}$;
(iv) $Ob_{\mathbf{A}}(\mathcal{B}) \not\models C(x)$ for each $\mathbf{A}C(x) \in N_{\mathcal{B}}$.

If there is an open branch $\mathcal{B}$ for Σ, then there is an $\mathcal{ALCK}_{\mathcal{NF}}$ structure $(\mathcal{M}, \mathcal{N})$ that satisfies Σ.

Definition 20 (Pre-preferred Branch) *A branch $\mathcal{B}$ is pre-preferred if all of the following conditions hold:*

(i) $\mathcal{B}$ is completed and open;
(ii) $Ob_{\mathbf{K}}(\mathcal{B}) \models Ob_{\mathbf{A}}(\mathcal{B})$;
(iii) $Ob_{\mathbf{K}}(\mathcal{B}) \not\models C(x)$ for each $\mathbf{A}C(x) \in N_{\mathcal{B}}$.

If there is a pre-preferred branch $\mathcal{B}$ for Σ, then there is an $\mathcal{ALCK}_{\mathcal{NF}}$ structure $(\mathcal{M}, \mathcal{M})$ that satisfies Σ.

The minimality condition below checks whether the set of interpretations $\mathcal{M} = \{\mathcal{I} \mid \mathcal{I} \models Ob_{\mathbf{K}}(\mathcal{B})\}$ represents a model for Σ. The minimality condition checks the minimality of a branch up to the renaming of individuals introduced by the $\exists$-rule.

For a branch $\mathcal{B}$ and an injection f, the branch obtained from $\mathcal{B}$ by replacing each occurrence of individual x by $f(x)$ for each $x \in \mathcal{O}_\Sigma \setminus \mathcal{O}_{\mathcal{B}}$, is called a *renamed branch* of $\mathcal{B}$ and denoted $f(\mathcal{B})$.

Definition 21 (Minimality Condition) *Let $\mathcal{B}$ be a completed branch for Σ. $\mathcal{B}$ satisfies the minimality condition if there does not exist a completed and open branch $\mathcal{B}'$ for Σ and an injection $f : \mathcal{O}_{\mathcal{B}'} \backslash \mathcal{O}_\Sigma \to \mathcal{O}_{\mathcal{B}} \backslash \mathcal{O}_\Sigma$ such that $|\mathcal{O}_{\mathcal{B}'}| \leq |\mathcal{O}_{\mathcal{B}}|$ and all of the following conditions hold:*

(i) $Ob_{\mathbf{K}}(\mathcal{B}) \models Ob_{\mathbf{K}}(f(\mathcal{B}'))$;
(ii) $Ob_{\mathbf{K}}(f(\mathcal{B}')) \not\models Ob_{\mathbf{K}}(\mathcal{B})$;
(iii) $Ob_{\mathbf{K}}(\mathcal{B}) \models Ob_{\mathbf{A}}(f(\mathcal{B}'))$;
(iv) $Ob_{\mathbf{K}}(\mathcal{B}) \not\models C(x)$ for each $\mathbf{A}C(x) \in N_{f(\mathcal{B}')}$.

If minimality condition does not hold, there exists a structure $(\mathcal{M}', \mathcal{M})$ such that $\mathcal{M}' \supset \mathcal{M}$ and $(\mathcal{M}', \mathcal{M})$ satisfies Σ. This implies that $\mathcal{M}$ is not a model for Σ.

For the tableaux algorithm with this minimality condition to be complete [3], KBs must be restricted according to the following definition.

Definition 22 (Minimality-proper KB) *An $\mathcal{ALCK}_{\mathcal{NF}}$ KB is minimality-proper if it satisfies the following condition: for each $\mathbf{K}C \sqsubseteq D \in \Gamma$ and each $D(x) \in \mathcal{A}$*

(i) D does not contain a subconcept of the form $\forall\mathbf{A}R.E$; and
(ii) D does not contain a subconcept of the form $\exists\mathbf{A}R.E$ in a disjunction.

For example, $\mathbf{K}C \sqsubseteq \mathbf{K}D \sqcup \exists\mathbf{A}R.C$ is not minimality-proper, while $\mathbf{K}C \sqsubseteq \exists\mathbf{A}R.C$ is.

The minimality-proper condition prevents $\forall\mathbf{A}R.C(x)$ to appear in $\mathcal{B}'$ of the minimality condition 21. The fact the branch $\mathcal{B}'$ is open implies that $(\mathcal{M}', \mathcal{M}'') \models \Sigma$, where $\mathcal{M}' = \{\mathcal{I} \mid \mathcal{I} \models Ob_{\mathbf{K}}(\mathcal{B}'))$ and $\mathcal{M}'' = \{\mathcal{I} \mid \mathcal{I} \models Ob_{\mathbf{A}}(\mathcal{B}'))$. The purpose of conditions (iii) and (iv) in Definition 21 is to imply that $\mathcal{M} = \{\mathcal{I} \mid \mathcal{I} \models Ob_{\mathbf{K}}(\mathcal{B}))$ can replace $\mathcal{M}''$ to ensure that $(\mathcal{M}', \mathcal{M}) \models \Sigma$. This purpose would not be achieved if there were assertions of the form $\forall\mathbf{A}R.C(x) \in \mathcal{B}'$ because there could exist $R(x, y)$ such that $Ob_{\mathbf{K}}(\mathcal{B}) \models R(x, y)$ and $Ob_{\mathbf{A}}(f(\mathcal{B}')) \not\models R(x, y)$, which does not meet the requirement of "assuming as much as we know from $\mathcal{B}$". Because of the tableaux expansion rules, the branch $\mathcal{B}'$ does not contain $\mathbf{A}R(x, y), C(y)$ and all the assertions obtained from $C(y)$. For analogous reason the minimality-proper KBs forbid $\exists\mathbf{A}R.E$ in disjunctions. We prove later that for the minimality-proper KBs, the minimality check is correct.

Definition 23 (Preferred Branch) *A branch $\mathcal{B}$ is preferred if all of the following conditions hold:*

(i) $\mathcal{B}$ is pre-preferred;
(ii) $\mathcal{B}$ satisfies the minimality condition.

If there is a preferred branch $\mathcal{B}$ for Σ, then there is an $\mathcal{ALCK}_{\mathcal{NF}}$ structure $(\mathcal{M}, \mathcal{M})$ that satisfies Σ and for each set of interpretations $\mathcal{M}'$, if $\mathcal{M}' \supset \mathcal{M}$ then $(\mathcal{M}', \mathcal{M})$ does not satisfy Σ. We conclude that $\mathcal{B}$ is a model for Σ.

Lemma 24 (Termination) *Let Σ be a subjectively quantified and minimality-proper $\mathcal{ALCK_{NF}}$ KB. Then the tableaux algorithm for checking satisfiability of Σ always terminates.*

To state completeness we must define what it means for a branch to represent a model. Let $\mathcal{B}$ be a completed branch for Σ and $\mathcal{M}$ a model for Σ. We define that $\mathcal{B}$ *represents* $\mathcal{M}$ if $\mathcal{B}$ is preferred and there exists an injection $f : \mathcal{O_B} \setminus \mathcal{O_\Sigma} \to \Delta \setminus \mathcal{O_\Sigma}$ such that $\mathcal{M} = \{\mathcal{I} \mid \mathcal{I} \models Ob_{\mathbf{K}}(f(\mathcal{B}))\}$ holds.

Theorem 25 (Soundness and completeness) *Let Σ be a subjectively quantified and minimality-proper $\mathcal{ALCK_{NF}}$ KB. Then Σ is satisfiable if and only if there exists a preferred branch $\mathcal{B}$ of the tableaux for Σ.*

If $\mathcal{B}$ is a preferred branch for Σ, then $\mathcal{M} = \{\mathcal{I} \mid \mathcal{I} \models Ob_{\mathbf{K}}(\mathcal{B})\}$ is a model for Σ. If $\mathcal{M}$ is a model for Σ, then there exists a completed branch $\mathcal{B}$ for Σ that represents $\mathcal{M}$.

Theorem 26 (Decidability) *Let Σ be a subjectively quantified and minimality-proper $\mathcal{ALCK_{NF}}$ KB. Then the satisfiability problem of Σ is decidable.*

4 Conclusion

For reasoning in $\mathcal{ALCK_{NF}}$ we presented a tableaux algorithm with blocking. The blocking is employed to deal with modal part of MKNF-DL theory. This technique allows for reasoning about KBs which are both subjectively quantified and minimality-proper. This is a larger class of KBs than in previous approaches, which further restricted to simple KBs.

References

1. Donini, F. M., Nardi, D., Rosati, R.: Description logics of minimal knowledge and negation as failure. *ACM Trans. Comput. Logic*, 3(2):177–225, 2002.
2. Gottlob, G.: Complexity results for nonmonotonic logics. *Journal of Logic and Computation*, (2):397–425, 1992.
3. Ke, P., Sattler, U.: Next steps for description logics of minimal knowledge and negation as failure. In Franz Baader, Carsten Lutz, and Boris Motik, editors, *Description Logics*, volume 353 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2008.
4. Malenko, J.: Reasoning in decription logics. Charles University in Prague, dissertation thesis, 2012.
5. Marek, V. W., Truszczynski, M.: Nonmonotonic logic: context-dependent reasoning. Springer-Verlag, Berlin, 1993.
6. Niemela, I.: Autoepistemic logic as a unified basis for nonmonotonic reasoning. 1993.