A Preferential Tableaux Calculus for Circumscriptive $\mathcal{ALCO}$ *

Stephan Grimm[1], Pascal Hitzler[2]

[1] FZI Research Center for Information Technologies, Univ. of Karlsruhe, Germany
[2] Institute AIFB, University of Karlsruhe, Germany

Abstract. Nonmonotonic extensions of description logics (DLs) allow for default and local closed-world reasoning and are an acknowledged desired feature for applications, e.g. in the Semantic Web. A recent approach to such an extension is based on McCarthy's circumscription, which rests on the principle of minimising the extension of selected predicates to close off dedicated parts of a domain model. While decidability and complexity results have been established in the literature, no practical algorithmisation for circumscriptive DLs has been proposed so far. In this paper, we present a tableaux calculus that can be used as a decision procedure for concept satisfiability with respect to concept-circumscribed $\mathcal{ALCO}$ knowledge bases. The calculus builds on existing tableaux for classical DLs, extended by the notion of a preference clash to detect the non-minimality of constructed models.

1 Introduction

Modern description logics (DLs) [8] are formalisations of semantic networks and frame-based knowledge representation systems that build on classical logic and are the foundation of the W3C Web Ontology Language OWL [16]. However, to also capture non-classical features, such as default and local closed-world reasoning, nonmonotonic extensions to DLs have been investigated. While in the past such extensions were primarily devised using autoepistemic operators [4, 12, 10] and default inclusions [1], a recent proposal [2] is to extend DLs by circumscription and to perform nonmonotonic reasoning on circumscribed DL knowledge bases. In circumscription, the extension of selected predicates – i.e. concepts or roles in the DL case – can be explicitly minimised to close off dedicated parts of a domain model, resulting in a default reasoning behaviour. In contrast to the former approaches, nonmonotonic reasoning in circumscriptive DLs also applies to "unknown individuals" that are not explicitly mentioned in a knowledge base, but whose existence is guaranteed due to existential quantification [7].

The proposal in [2] presents a semantics for circumscriptive DLs together with decidability and complexity results, in particular for fragments of the logic $\mathcal{ALCQIO}$. However, a practical algorithmisation for reasoning in circumscriptive DLs has not been addressed so far. In this paper, we present an algorithm that builds on existing DL tableaux methods for guided model construction. In

* This work is partially supported by the Deutsche Forschungsgemeinschaft (DFG) under the ReaSem project and by the the German Federal Ministry of Education and Research (BMBF) under the Theseus project, `http: //theseus-programm.de`.

particular, we present a tableaux calculus that supports reasoning with concept-circumscribed knowledge bases in the logic $\mathcal{ALCO}$. We focus on the reasoning task of concept satisfiability, which is motivated by an application of nonmonotonic reasoning in a Semantic Web setting, described in [7]. While typical examples in the circumscription literature deal with defeasible conclusions of circumscriptive abnormality theories, in this setting we use minimisation of concepts to realise a local closed-world assumption for the matchmaking of semantically annotated resources.

The reason for our choice of $\mathcal{ALCO}$ as the underlying DL is twofold. First, we want to present the circumscriptive extensions for the simplest expressive DL $\mathcal{ALC}$ for sake of a clear and concise description of the tableaux modifications. Second, there is the necessity to deal with nominals within the calculus in order to keep track of extensions of minimised concepts, so we include $\mathcal{O}$.

The basic idea behind our calculus is to detect the non-minimality of candidate models, produced by a tableaux procedure for classical DLs, via the notion of a preference clash, and based on the construction of a classical DL knowledge base that has a model if and only if the original candidate model produced is not minimal. This check can be realised by reasoning in classical DLs with nominals. We formally prove this calculus to be sound and complete. A similar idea has been applied in [13] for circumscriptive reasoning in first-order logic. However, that calculus does not directly yield a decision procedure for reasoning with DLs as it is only decidable if function symbols are disallowed, which correspond to existential restrictions in DLs.

The paper is structured as follows. In Section 2 we recall circumscriptive DLs from [2] for the case of $\mathcal{ALCO}$. In Section 3, we present our tableaux calculus and prove it to be a decision procedure for circumscriptive concept satisfiability. We conclude in Section 4. Due to the page limit, we could not include all proofs in full. They can be found in [6].

2 Description Logics and Circumscription

Description Logics (DLs) [8] are typically fragments of first-order predicate logic that provide a well-studied formalisation for knowledge representation systems. Circumscription [11], on the other hand, is an approach to nonmonotonic reasoning based on the explicit minimisation of selected predicates. In this section, we present the description logic $\mathcal{ALCO}$ extended with circumscription according to [2], which allows for nonmonotonic reasoning with DL knowledge bases.

2.1 Circumscriptive $\mathcal{ALCO}$

The basic elements to represent knowledge in DLs are *individuals* that represent objects in a domain of discourse, *concepts* that group together individuals with common properties, and *roles* that put individuals in relation. The countably infinite sets N_I, N_C and N_r of individual names, concept names and role names, respectively, form the basis to construct the syntactic elements of $\mathcal{ALCO}$ according to the following grammar, in which $A \in \mathsf{N}_C$ denotes an atomic concept, $C_{(i)}$

denote complex concepts, $r \in \mathsf{N}_r$ denotes a role and $a_i \in \mathsf{N}_I$ denote individuals.

$$C_{(i)} \longrightarrow \bot \mid \top \mid A \mid \neg C \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \exists\, r\,.C \mid \forall\, r\,.C \mid \{a_1, \ldots, a_n\}$$

The *negation normal form* of a concept C, which we denote by $\|C\|$, is obtained from pushing negation symbols $\neg$ into concept expression to occur in front of atomic concepts only, as described in [15].

The semantics of the syntactic elements of $\mathcal{ALCO}$ is defined in terms of an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ with a non-empty set $\Delta^{\mathcal{I}}$ as the *domain* and an *interpretation function* $\cdot^{\mathcal{I}}$ that maps each individual $a \in \mathsf{N}_I$ to a distinct element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ and that interprets (possibly) complex concepts and roles as follows.

$$\begin{aligned}
\top^{\mathcal{I}} = \Delta^{\mathcal{I}}\ ,\ \bot^{\mathcal{I}} &= \emptyset\ ,\ A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}\ ,\ r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \\
(C_1 \sqcap C_2)^{\mathcal{I}} &= C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}} \\
(C_1 \sqcup C_2)^{\mathcal{I}} &= C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}} \\
(\neg C)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \\
(\forall\, r\,.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \forall y.(x, y) \in r^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\} \\
(\exists\, r\,.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \exists y.(x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\} \\
(\{a_1, \ldots, a_n\})^{\mathcal{I}} &= \{a_1^{\mathcal{I}}, \ldots, a_n^{\mathcal{I}}\}
\end{aligned}$$

Notice that we assume unique names for individuals, i.e. $a_1^{\mathcal{I}} \neq a_2^{\mathcal{I}}$ for any interpretation $\mathcal{I}$ and any pair $a_1, a_2 \in \mathsf{N}_I$.

An $\mathcal{ALCO}$ knowledge base *KB* is a finite set of *axioms* formed by concepts, roles and individuals. A *concept assertion* is an axiom of the form $C(a)$ that assigns membership of an individual a to a concept C. A *role assertion* is an axiom of the form $r(a_1, a_2)$ that assigns a directed relation between two individuals a_1, a_2 by the role r. A *concept inclusion* is an axiom of the form $C_1 \sqsubseteq C_2$ that states the subsumption of the concept C_1 by the concept C_2, while a *concept equivalence* axiom $C_1 \equiv C_2$ is a shortcut for two inclusions $C_1 \sqsubseteq C_2$ and $C_2 \sqsubseteq C_1$. An interpretation $\mathcal{I}$ satisfies a concept assertion $C(a)$ if $a^{\mathcal{I}} \in C^{\mathcal{I}}$, a role assertion $r(a_1, a_2)$ if $(a_1^{\mathcal{I}}, a_2^{\mathcal{I}}) \in r^{\mathcal{I}}$, a concept inclusion $C_1 \sqsubseteq C_2$ if $C_1^{\mathcal{I}} \subseteq C_2^{\mathcal{I}}$ and a concept equivalence $C_1 \equiv C_2$ if $C_1^{\mathcal{I}} = C_2^{\mathcal{I}}$. An interpretation that satisfies all axioms of a knowledge base *KB* is called a *model* of *KB*. A concept C is called *satisfiable with respect to KB* if *KB* has a model in which $C^{\mathcal{I}} \neq \emptyset$ holds.

We now turn to the circumscription part of the formalism, that allows for nonmonotonic reasoning by explicit minimisation of selected $\mathcal{ALCO}$ concepts. We adopt a slightly simplified form of the circumscriptive DLs presented in [2] by restricting our formalism to parallel concept circumscription (without prioritisation among minimised concepts). For this purpose we define the notion of a *circumscription pattern* as follows.

Definition 1 (circumscription pattern, $<_{\mathsf{CP}}$). *A* circumscription pattern[3] *CP is a tuple (M, F, V) of sets of atomic concepts called the* minimised*,* fixed *and*

[3] The notion of circumscription pattern introduced in [2] is more general and allows the sets M, F and V to also contain roles. There, a circumscription pattern according to Definition 1 is called a concept circumscription pattern. However, in the general case role circumscription leads to undecidability, which was also shown in [2]. As our calculus does not allow for role circumscription, we use the term circumscription pattern to denote a concept circumscription pattern in the sense of [2].

varying *concepts. Based on* CP, *a preference relation on interpretations is defined by setting* $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$ *if and only if the following conditions hold:*

(i) $\Delta^{\mathcal{J}} = \Delta^{\mathcal{I}}$ *and* $a^{\mathcal{J}} = a^{\mathcal{I}}$ *for all* $a^{\mathcal{J}} \in \Delta^{\mathcal{J}}$
(ii) $\bar{A}^{\mathcal{J}} = \bar{A}^{\mathcal{I}}$ *for all* $\bar{A} \in F$
(iii) $\tilde{A}^{\mathcal{J}} \subseteq \tilde{A}^{\mathcal{I}}$ *for all* $\tilde{A} \in M$
(iv) there is an $\tilde{A} \in M$ *such that* $\tilde{A}^{\mathcal{J}} \subset \tilde{A}^{\mathcal{I}}$

For nonmonotonic reasoning, a classical $\mathcal{ALCO}$ knowledge base is circumscribed with a circumscription pattern and reasoning is performed by means of the resulting *circumscribed knowledge base*, defined as follows.

Definition 2 (circumscribed knowledge base). *A* circumscribed knowledge base *circ*$_{\mathsf{CP}}(\mathit{KB})$ *is a knowledge base* KB *together with a circumscription pattern* $\mathsf{CP} = (M, F, V)$, *such that the sets* M, F *and* V *partition the atomic concepts that occur in* KB. *An interpretation* $\mathcal{I}$ *is a model of circ*$_{\mathsf{CP}}(\mathit{KB})$ *if* $\mathcal{I}$ *is a model of* KB *and there exists no model* $\mathcal{J}$ *of* KB *with* $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$.

The intuition behind the preference relation is to identify interpretations that are "smaller" in the extensions of minimised concepts than others, to select only the "smallest" ones as the *preferred* models. Fixed concepts can be used to restrict this selection and to prevent certain models from being preferred.

2.2 Reasoning with Circumscribed Knowledge Bases

The typical DL reasoning tasks are defined as expected (see [2]) with respect to the models of a circumscribed knowledge base $\mathrm{circ}_{\mathsf{CP}}(\mathit{KB})$, which are just the preferred models of KB with respect to CP. For our calculus, we focus on concept satisfiability, which we define next. Other reasoning tasks can be reduced to concept satisfiability, as described in [2].

Definition 3 (circumscriptive concept satisfiability). *A concept* C *is* satisfiable with respect to a circumscribed knowledge base $\mathrm{circ}_{\mathsf{CP}}(\mathit{KB})$ *if some model* $\mathcal{I}$ *of circ*$_{\mathsf{CP}}(\mathit{KB})$ *satisfies* $C^{\mathcal{I}} \neq \emptyset$.

Observe that in classical DLs an atomic concept A is satisfiable with respect to a knowledge base KB "by default" if there is no evidence for its unsatisfiability in KB, i.e. any A is satisfiable with respect to the empty knowledge base. Now suppose that A is a minimised concept in a circumscription pattern CP by which KB is circumscribed. Then, A is unsatisfiable with respect to $\mathrm{circ}_{\mathsf{CP}}(\mathit{KB})$ for $\mathit{KB} = \emptyset$. Only if we explicitly assure that the extension of A is non-empty, e.g. by setting $\mathit{KB} = \{A(a)\}$, A becomes satisfiable.

A known result in circumscription is that there is a close relation between fixed and minimised predicates. Namely, fixed predicates can be simulated by minimising them together with their complements. In case of concept circumscription this is achieved by introducing additional concept names and respective equivalence axioms, as reflected by the following proposition (see [2, 3, 6]).

Proposition 1 (simulation of concept fixation). *Let C be a concept, let KB be a knowledge base and let $\mathsf{CP} = (M, F, V)$ be a circumscription pattern with $F = \{\bar{A}_1, \ldots, \bar{A}_n\}$. Furthermore, let*

$$\mathit{KB}' = \mathit{KB} \cup \{\tilde{A}_i \equiv \neg\bar{A}_i \mid 1 \leq i \leq n\}$$

$$\text{and let} \quad \mathsf{CP}' = (M \cup \{\tilde{A}_1, \ldots, \tilde{A}_n, \bar{A}_1, \ldots, \bar{A}_n\}, \emptyset, V) \ ,$$

where $\tilde{A}_1, \ldots, \tilde{A}_n$ are atomic concepts that do not occur in KB, CP or C. Then, C is satisfiable with respect to $\mathit{circ}_{\mathsf{CP}}(\mathit{KB})$ if and only if it is satisfiable with respect to $\mathit{circ}_{\mathsf{CP}'}(\mathit{KB}')$.

To illustrate the reasoning task of checking concept satisfiability with respect to circumscribed knowledge bases we present the following example.

Example 1. The following knowledge base describes species of the arctic sea.

$$\mathit{KB}_1 = \{ \ \mathit{Bears}(\mathit{PolarBear}), \ \neg\mathit{Bears}(\mathit{BlueWhale}), \ \mathit{EndangeredSpecies}(\mathit{BlueWhale}) \ \}$$

According to KB_1, the polar bear is a kind of bear, while the blue whale is not. Moreover, the blue whale is explicitly listed to be an endangered species, while the polar bear does not occur on this list. The following circumscription pattern allows to "switch off" the open-world assumption for the list of endangered species by minimising the extension of the concept *EndangeredSpecies*.

$$\mathsf{CP} = (M = \{\mathit{EndangeredSpecies}\}, F = \emptyset, V = \{\mathit{Bears}\})$$

The concept $\mathit{Bears} \sqcap \mathit{EndangeredSpecies}$ is unsatisfiable with respect to the circumscribed knowledge base $\mathrm{circ}_{\mathsf{CP}}(\mathit{KB}_1)$, reflecting that there cannot be an individual that is both an endangered species and a kind of bear. The only endangered species in the preferred models of KB_1 is the blue whale, which is explicitly said to be no kind of bear.

Recently, however, the polar bear unfortunately had to be included in the list of endangered species, which is reflected by the following update of KB_1.

$$\mathit{KB}_2 = \mathit{KB}_1 \cup \{ \ \mathit{EndangeredSpecies}(\mathit{PolarBear}) \ \}$$

With respect to $\mathrm{circ}_{\mathsf{CP}}(\mathit{KB}_2)$, the concept $\mathit{Bears} \sqcap \mathit{EndangeredSpecies}$ is satisfiable, as the polar bear is a kind of bear and at the same time an endangered species in the preferred models of KB_2.

Instead of using a concept assertion for the explicitly mentioned individual *PolarBear*, we could alternatively update KB_1 by introducing an existentially quantified object through an inclusion axiom stating that the arctic sea is a habitat for an endangered bear species, as follows.

$$\mathit{KB}_3 = \mathit{KB}_1 \cup \{ \ \exists\, \mathit{isHabitatFor}.(\mathit{Bears} \sqcap \mathit{EndangeredSpecies})(\mathit{ArcticSea}) \ \}$$

The concept $\mathit{Bears} \sqcap \mathit{EndangeredSpecies}$ is also satisfiable with respect to $\mathrm{circ}_{\mathsf{CP}}(\mathit{KB}_3)$. Observe that in any preferred model of KB_3 the extension of *EndangeredSpecies* contains an unknown individual whose existence is propagated from the known individual *ArcticSea* via the role *isHabitatFor*. Alternative approaches to non-monotonic reasoning in DL, such as [5, 1], typically treat unknown objects differently and do not allow for this kind of reasoning.

3 Tableaux Calculus for Circumscriptive $\mathcal{ALCO}$

In this section, we introduce a preferential tableaux calculus that decides the satisfiability of a concept with respect to a circumscribed knowledge base. We build on the notion of constraint systems, which map to tableaux branches in tableaux calculi, and we keep the presentation similar to that in [4].

3.1 Constraint Systems and their Solvability

In addition to the alphabet of individuals N_I, we introduce a set N_V of variable symbols. We denote elements of N_I by a, elements of N_V by x and elements of $\mathsf{N}_I \cup \mathsf{N}_V$ by o, all possibly with an index. A *constraint* is a syntactic entity of one of the forms $o : C$ or $(o_1, o_2) : r$ or $\forall x.x : C$, where C is an $\mathcal{ALCO}$ concept, r is a role and the o's are objects in $\mathsf{N}_I \cup \mathsf{N}_V$. A *constraint system*, denoted by S, is a finite set of constraints. By N_I^S we denote the individuals and by N_V^S the variables that occur in a constraint system S.

Given an interpretation $\mathcal{I}$, we define an $\mathcal{I}$-*assignment* as a function $\alpha^\mathcal{I} : \mathsf{N}_I \cup \mathsf{N}_V \mapsto \Delta^\mathcal{I}$, that maps every variable of N_V to an element of $\Delta^\mathcal{I}$ and every individual a to $a^\mathcal{I}$, i.e. $\alpha^\mathcal{I}(a) = a^\mathcal{I}$ for all $a \in \mathsf{N}_I$.

A pair $(\mathcal{I}, \alpha^\mathcal{I})$ of an interpretation $\mathcal{I}$ and an $\mathcal{I}$-assignment $\alpha^\mathcal{I}$ *satisfies* a constraint $o : C$ if $\alpha^\mathcal{I}(o) \in C^\mathcal{I}$, a constraint $(o_1, o_2) : r$ if $(\alpha^\mathcal{I}(o_1), \alpha^\mathcal{I}(o_2)) \in r^\mathcal{I}$ and a constraint $\forall x.x : C$ if $C^\mathcal{I} = \Delta^\mathcal{I}$. A *solution* for a constraint system S is a pair $(\mathcal{I}, \alpha^\mathcal{I})$ of an interpretation $\mathcal{I}$ and an $\mathcal{I}$-assignment $\alpha^\mathcal{I}$ that satisfies all constraints in S.

We denote by $S[o_1/o_2]$ the constraint system that is obtained by replacing any occurrence of object o_1 by object o_2 in every constraint in S. Furthermore, we define the constraint system S_{KB} to be obtained from an $\mathcal{ALCO}$ knowledge base KB by including one constraint of the form $a : \|C\|$ for each concept assertion $C(a) \in \mathit{KB}$, one constraint $(a_1, a_2) : r$ for each role assertion $r(a_1, a_1) \in \mathit{KB}$ and one constraint $\forall x.x : \|\neg C_1 \sqcup C_2\|$ for each concept inclusion $C_1 \sqsubseteq C_2 \in \mathit{KB}$, such that S_{KB} captures all the information in KB.

To ensure termination of our calculus in the presence of general inclusion axioms, we need to introduce the notion of blocking (see e.g. [9]). Given a constraint system S and $S^* \subseteq S$, we say that an object o_1 is a *direct predecessor* of an object o_2, if S^* contains a role constraint $(o_1, o_2) : r$ for some role r. We denote by *predecessor* the transitive closure in S^* of the direct predecessor relation. Moreover, we say that, in a constraint system S with $S^* \subseteq S$, an object o_2 is *blocked by* an object o_1 if o_1 is a predecessor of o_2 and if $\{C \mid o_2 : C \in S\} \subseteq \{C \mid o_1 : C \in S\}$ holds. The set S^* is generated by the algorithm; it is used to control which role constraints in S shall be taken into consideration for blocking.

Due to the analogy between a constraint system and a knowledge base the following Lemma holds.

Lemma 1. *Let KB be an $\mathcal{ALCO}$ knowledge base, S be a constraint system with $S_{\mathit{KB}} \subseteq S$ and $\mathcal{I}$ be an interpretation. If $\mathcal{I}$ is a model of KB then, for any $\mathcal{I}$-assignment $\alpha^\mathcal{I}$, $(\mathcal{I}, \alpha^\mathcal{I})$ is a solution for S_{KB}. Furthermore, for any solution $(\mathcal{I}, \alpha^\mathcal{I})$ for S, $\mathcal{I}$ is a model of KB.*

Our calculus is based on finding a solution for constraint systems the interpretation of which is a preferred model of an initial knowledge base with respect to a circumscription pattern. For this purpose we define the notion of solvability.

Definition 4 (CP-solvability). *A constraint system S is* CP-solvable with respect to *KB if there is a model $\mathcal{I}$ of KB and an $\mathcal{I}$-assignment $\alpha^{\mathcal{I}}$ such that $(\mathcal{I}, \alpha^{\mathcal{I}})$ is a solution for S and there is no model $\mathcal{J}$ of KB with $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$.*

By the next proposition, we reduce circumscriptive concept satisfiability to checking a constraint system for its solvability.

Proposition 2 (satisfiability reduction). *Let KB be a knowledge base, CP be a circumscription pattern and C be a concept. C is satisfiable with respect to $\mathit{circ}_{\mathsf{CP}}(\mathit{KB})$ if and only if $S_{\mathit{KB}} \cup \{x : C\}$ is CP-solvable with respect to KB.*

Proof. $\Rightarrow$: Since C is satisfiable with respect to $\mathrm{circ}_{\mathsf{CP}}(\mathit{KB})$, there is a model $\mathcal{I}$ of $\mathrm{circ}_{\mathsf{CP}}(\mathit{KB})$ in which $C^{\mathcal{I}}$ is nonempty. Let a be an individual with $a^{\mathcal{I}} \in C^{\mathcal{I}}$. Since $\mathcal{I}$ is also a model of KB and due to Lemma 1, $(\mathcal{I}, \alpha^{\mathcal{I}})$ is a solution for S_{KB} for any $\mathcal{I}$-assignment $\alpha^{\mathcal{I}}$. Let $\alpha^{\mathcal{I}}_{x,a}$ be an $\mathcal{I}$-assignment with $\alpha^{\mathcal{I}}_{x,a}(x) = a^{\mathcal{I}}$. Then, $(\mathcal{I}, \alpha^{\mathcal{I}}_{x,a})$ satisfies, besides the constraints in S_{KB}, also the constraint $x : C$, because of $\alpha^{\mathcal{I}}_{x,a}(x) \in C^{\mathcal{I}}$, and is therefore a solution for $S_{\mathit{KB}} \cup \{x : C\}$. Since there is no other model $\mathcal{J}$ of KB with $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$, $S_{\mathit{KB}} \cup \{x : C\}$ is CP-solvable with respect to KB.
$\Leftarrow$: Since $S_{\mathit{KB}} \cup \{x : C\}$ is CP-solvable with respect to KB, there is a model $\mathcal{I}$ of KB and an $\mathcal{I}$-assignment $\alpha^{\mathcal{I}}$ such that $(\mathcal{I}, \alpha^{\mathcal{I}})$ is a solution for $S_{\mathit{KB}} \cup \{x : C\}$. Moreover, there exists an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ with $\alpha^{\mathcal{I}}(x) = a^{\mathcal{I}} \in C^{\mathcal{I}}$ because $(\mathcal{I}, \alpha^{\mathcal{I}})$ satisfies the constraint $x : C$. By definition of CP-solvability, there is no model $\mathcal{J}$ of KB with $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$, and thus, $\mathcal{I}$ is a model of $\mathrm{circ}_{\mathsf{CP}}(\mathit{KB})$ in which $C^{\mathcal{I}}$ is non-empty. Hence, C is satisfiable with respect to $\mathrm{circ}_{\mathsf{CP}}(\mathit{KB})$. □

3.2 Tableaux Expansion Rules

Constraint systems are manipulated by tableaux expansion rules, which decompose the structure of complex logical constructs or replace variables by concrete individuals. By expanding a constraint system with the resulting constraints, our calculus tries to build a model for the initial knowledge base that is represented by the constraint system. To decide the satisfiability of a concept C with respect to a circumscribed knowledge base $\mathrm{circ}_{\mathsf{CP}}(\mathit{KB})$ according to Proposition 2, we initialise the calculus with the constraint system $S_{\mathit{KB}} \cup \{x : C\}$ and $S^* = S$. The algorithm exhaustively performs the tableau rules given in Table 1, however the $\longrightarrow_{<_{\mathsf{CP}}}$-rule must not be applied if any of the other rules is applicable, i.e. the $\longrightarrow_{<_{\mathsf{CP}}}$-rule has a lower precedence than the other rules. The notion of predecessor is evaluated with respect to S^*. Without loss of generality, we assume all fixed predicates to be simulated according to Proposition 1, and thus, the set F in CP to be empty without loss of generality.

Observe that the rules are parametric with respect to KB and CP. The rules $\longrightarrow_{\forall_x}$, $\longrightarrow_{\sqcap}$, $\longrightarrow_{\exists}$ and $\longrightarrow_{\forall}$ are *deterministic* and their application yields a

Table 1. Tableau Expansion Rules for Circumscriptive $\mathcal{ALCO}$. The $\longrightarrow_{<_{\mathsf{CP}}}$-rule must not be executed if any of the other rules is applicable. Blocking is evaluated with respect to S^*.

$\longrightarrow_{\forall_x}$	:	**if** $\forall x.x : C \in S$ and $o : C \notin S$ for some $o \in \mathsf{N}_I^S \cup \mathsf{N}_V^S$ **then** S$\leftarrow S \cup \{o : C\}$
$\longrightarrow_{\sqcap}$	:	**if** $o : C_1 \sqcap C_2 \in S$ and $\{o : C_1, o : C_2\} \not\subseteq S$ **then** S$\leftarrow S \cup \{o : C_1, o : C_2\}$
$\longrightarrow_{\sqcup}$	:	**if** $o : C_1 \sqcup C_2 \in S$ and $\{o : C_1, o : C_2\} \cap S = \emptyset$ **then** $S \leftarrow S \cup \{o : C_1\}$ or $S \leftarrow \{o : C_2\}$
$\longrightarrow_{\exists}$	:	**if** $o_1 : \exists\, r\,.C \in S$ and $\{(o_1, o_2) : r, o_2 : C\} \not\subseteq S$ and o_1 is not blocked **then** $S \leftarrow S \cup \{(o_1, x) : r, x : C\},$ with x a new variable and $(o_1, x) : r$ is added to S^*.
$\longrightarrow_{\forall}$	:	**if** $o_1 : \forall\, r\,.C \in S$ and $(o_1, o_2) : r \in S$ and $o_2 : C \notin S$ **then** $S \leftarrow S \cup \{o_2 : C\}$
$\longrightarrow_{\mathcal{O}}$	:	**if** $x : \{a_1, \ldots, a_k\} \in S$ **then** $S \leftarrow S[x/a_i]$ for any $i \in \{1, \ldots, k\} \subset \mathbb{N}$ and all $(o, x) : r$ are removed from S^*
$\longrightarrow_{<_{\mathsf{CP}}}$	:	**if** $x : \tilde{A} \in S$ and $\tilde{A} \in M$ **then** $S \leftarrow S[x/a]$ for some $a \in \mathsf{N}_I^S \cup \{\iota\},$ with ι a new individual and $S^* \leftarrow S^*[x/\iota]$ if $a = \iota$

single constraint system. Contrarily, the rules $\longrightarrow_{\sqcup}$, $\longrightarrow_{\mathcal{O}}$ and $\longrightarrow_{<_{\mathsf{CP}}}$ are *non-deterministic*, meaning that they can be applied in multiple ways that yield different constraint systems. Any such non-deterministic choice produces a branching point for backtracking. In the $\longrightarrow_{\sqcup}$-rule, the disjunction leads to the choice of expanding on either of the disjuncts, while in the $\longrightarrow_{\mathcal{O}}$- and $\longrightarrow_{<_{\mathsf{CP}}}$-rules the presence of several individuals leads to a choice of selecting one for replacement of the variable x. Moreover, the $\longrightarrow_{<_{\mathsf{CP}}}$-rule introduces new individuals into the constraint system whenever ι is selected for replacement,[4] while the $\longrightarrow_{\exists}$-rule introduces new variables whenever an object lacks a role filler.

Definition 5 (completion). *A* completion *of a constraint system S with regard to* CP *and KB is any constraint system that results from the application of the algorithm to S, using* CP *and KB, and to which none of the rules is applicable.*

The algorithm finally leads to a completion of the initial constraint system that contains the exhaustive decomposition of complex constraints, which is established by the following lemma.

Lemma 2 (termination). *For any constraint system S, the algorithm always terminates, and yields a completion of S.*

[4] The idea of including a new individual ι as a representative for the infinitely many remaining objects in $\mathsf{N}_I \setminus \mathsf{N}_I^S$ in the domain is taken from [4].

Proof (Sketch). Note that the top part of Table 1 (without the $\longrightarrow_{<_{\mathsf{CP}}}$-rule) and corresponding algorithm coincides with that of [9] for $\mathcal{ALCO}$. In fact, the termination proof from [9], can easily be adapted to our setting.

Moreover, we establish the result that the tableaux expansion rules of our calculus preserve the solvability of constraint systems as follows.

Proposition 3 (solvability preservation). *Let KB be a knowledge base,* CP *be a circumscription pattern and* S, S' *be two constraint systems.*

1. *If* S' *results from* S *by application of a deterministic rule then* S *is* CP*-solvable with respect to KB if and only if* S' *is* CP*-solvable with respect to KB.*
2. *If* S' *results from* S *by application of a non-deterministic rule then* S *is* CP*-solvable with respect to KB if* S' *is* CP*-solvable with respect to KB. Furthermore, if* S *is* CP*-solvable with respect to KB and a non-deterministic rule applies to* S *then it can be applied in such a way that the resulting constraint system* S' *is also* CP*-solvable with respect to KB.*

Proof. The claim 1. for the rules $\longrightarrow_{\sqcap}$, $\longrightarrow_{\exists}$, $\longrightarrow_{\forall}$, $\longrightarrow_{\sqcup}$ and $\longrightarrow_{\mathcal{O}}$ follows from the results in [9]. (See also [6] for a full proof.) Therefore, we concentrate on the claim 2. for the $\longrightarrow_{<_{\mathsf{CP}}}$-rule.
$\Leftarrow$: Assume that S' is obtained from S by application of the $\longrightarrow_{<_{\mathsf{CP}}}$-rule and S' is CP-solvable with respect to *KB*. Let $(\mathcal{I}, \alpha^{\mathcal{I}})$ be a solution for S' such that $\mathcal{I}$ is a model of *KB* and there is no model $\mathcal{J}$ of *KB* with $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$. As the $\longrightarrow_{<_{\mathsf{CP}}}$-rule has been applied, $S' = S[x/a]$ for some individual $a \in \mathsf{N}_I$. As a solution for S', $(\mathcal{I}, \alpha^{\mathcal{I}})$ satisfies all the constraints in $S[x/a]$, in particular those in which x has been replaced by a. Let $\alpha^{\mathcal{I}}_{x,a}$ be the $\mathcal{I}$-assignment that coincides with $\alpha^{\mathcal{I}}$ except that $\alpha^{\mathcal{I}}_{x,a}(x) = a^{\mathcal{I}}$. Then, $(\mathcal{I}, \alpha^{\mathcal{I}}_{x,a})$ satisfies all the constraints in S in which x occurs, and since S and S' differ only by these, also all remaining constraints in S. Hence, $(\mathcal{I}, \alpha^{\mathcal{I}}_{x,a})$ is a solution for S, and since there is no model $\mathcal{J}$ of *KB* with $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$ by assumption, S is CP-solvable with respect to *KB*.
$\Rightarrow$: Assume that S' is obtained from S by application of the $\longrightarrow_{<_{\mathsf{CP}}}$-rule and that S is CP-solvable with respect to *KB*. Let $(\mathcal{I}, \alpha^{\mathcal{I}})$ be a solution for S such that $\mathcal{I}$ is a model of *KB* and there is no model $\mathcal{J}$ of *KB* with $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$. As the $\longrightarrow_{<_{\mathsf{CP}}}$-rule has been applied, S contains a constraint of the form $x : \tilde{A}$ with $\tilde{A} \in M$. As a solution for S, $(\mathcal{I}, \alpha^{\mathcal{I}})$ satisfies this constraint and there is some individual $a \in \mathsf{N}_I$ with $\alpha^{\mathcal{I}}(x) = a^{\mathcal{I}}$. We distinguish the two cases in which a) a is in N_I^S and b) a is a new individual not in N_I^S:

- a) In case $a \in \mathsf{N}_I^S$, a can be picked for the application of the $\longrightarrow_{<_{\mathsf{CP}}}$-rule and it directly follows that $(\mathcal{I}, \alpha^{\mathcal{I}})$ is a solution for the resulting constraint system $S' = S[x/a]$.
- b) In case $a \in \mathsf{N}_I \setminus \mathsf{N}_I^S$, $\iota \in \mathsf{N}_I \setminus \mathsf{N}_I^S$ can be picked for the application of the $\longrightarrow_{<_{\mathsf{CP}}}$-rule as a representative for any new individual. Then, $S[x/a]$ and $S[x/\iota]$ differ only by the naming of an individual new to S and are in this sense isomorphic[5]. Hence, as $(\mathcal{I}, \alpha^{\mathcal{I}})$ is a solution for $S[x/a]$ it is also a solution for the resulting constraint system $S[x/\iota] = S'$.

[5] See also the analogous argument in [4, Lemma 3.6].

Algorithm 1 Construct a knowledge base KB'.

Require: a constraint system S produced for an initial $\mathcal{ALCO}$ knowledge base KB circumscribed with a circumscription pattern $\mathsf{CP} = (M, F, V)$

$\mathit{KB}' \leftarrow \mathit{KB}$, $D \leftarrow \{\bot\}$
for all $\tilde{A} \in M_{\mathit{KB}}$ **do**
  $E_{\tilde{A}} := \{a \mid a : \tilde{A} \in S\}$
  if $\#E_{\tilde{A}} > 0$ **then**
    $\mathit{KB}' \leftarrow \mathit{KB}' \cup \{\tilde{A} \sqsubseteq \{a_1, \ldots, a_n\}\}$, $a_1, \ldots a_n \in E_{\tilde{A}}$
    $D \leftarrow D \cup \{\{a_1, \ldots, a_n\} \sqcap \neg\tilde{A}\}$, $a_1, \ldots a_n \in E_{\tilde{A}}$
  else
    $\mathit{KB}' \leftarrow \mathit{KB}' \cup \{\tilde{A} \sqsubseteq \bot\}$
  end if
end for
$\mathit{KB}' \leftarrow \mathit{KB}' \cup \{(\bigsqcup_{D_{\tilde{A}} \in D} D_{\tilde{A}})(\iota)\}$, with ι a new individual

Finally, since $(\mathcal{I}, \alpha^{\mathcal{I}})$ is a solution for S' and there is no model $\mathcal{J}$ of KB with $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$ by assumption, the $\longrightarrow_{<\mathsf{CP}}$-rule can be applied to S in such a way that S' is CP-solvable with respect to KB. □

3.3 Notions of Clash and Detection of Inconsistencies

Once a completion of an initial constraint system has been produced, its solvability can be verified by using the notion of a *clash*. In addition to the clashes defined in [4, 14], which represent obvious contradictions in a knowledge base, we introduce the notion of a *preference clash*, which reflects non-minimality of the respective model with regard to the preference relation $<_{\mathsf{CP}}$.

Definition 6 (Clashes). *Let S be a constraint system.*

S contains an inconsistency clash *if at least one of the following holds:*

(i) S contains a constraint of the form $o : \bot$.
(ii) S contains two constraints of the form $o : A$, $o : \neg A$.

S contains an individual clash *if at least one of the following holds:*

(iii) S contains a constraint of the form $a : \{a_1, \ldots, a_k\}$. with $a \neq a_i$ for all $i \in \{1, \ldots, k\} \subset \mathbb{N}$.
(vi) S contains a constraint of the form $a : \neg\{a_1, \ldots, a_k\}$. with $a = a_i$ for some $i \in \{1, \ldots, k\} \subset \mathbb{N}$.

S contains a preference clash*, parameterised with a circumscription pattern CP and an $\mathcal{ALCO}$ knowledge base KB, if the following condition holds:*

(v) the constraint system $S_{\mathit{KB}'}[\iota/x]$ has a completion, with regard to $\mathsf{CP}' = (\emptyset, \emptyset, F \cup M \cup V)$ and KB', that does neither contain an inconsistency clash nor an individual clash, while the $\mathcal{ALCO}$ knowledge base KB' is constructed according to Algorithm 1.

The idea behind the construction of KB' in Algorithm 1 is to freeze the instance situation for minimised concepts as asserted in the current constraint system perceived as reflecting some model $\mathcal{I}$ of the original knowledge base KB.

Then, KB' is constructed such that for any of its models $\mathcal{J}$ it holds that $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$, and thus, checking KB' for unsatisfiability verifies minimality of $\mathcal{I}$. By inclusion axioms for minimised concepts $\tilde{A}$ the conditions 2 and 3 of Definition 1 are assured to hold for each model of KB'. Moreover, by the disjunctive concept assertion condition 4 of Definition 1 is assured to hold, such that any model of KB' is actually "smaller" than $\mathcal{I}$ in some minimised concept, which is achieved by mapping the not uniquely named individual ι to one that already occurs in the extension of a minimised concept. Although in general we assume unique names in the formalism, the replacement of the new individual ι by the variable x within $S_{KB'}[\iota/x]$ in condition (v) of Definition 6 allows ι to be (indirectly) identified with some other individual.

We illustrate the detection of clashes in our calculus by means of an example.

Example 2. Consider the circumscribed knowledge base $\mathrm{circ}_{\mathsf{CP}}(KB)$ with the following knowledge base KB and circumscription pattern CP.

$$\begin{aligned} KB &= \{\ \neg \mathit{Bears}(\mathit{BlueWhale})\,,\ \mathit{EndangeredSpecies}(\mathit{BlueWhale})\ \} \\ \mathsf{CP} &= (M = \{\mathit{EndangeredSpecies}\}, F = \emptyset, V = \{\mathit{Bears}\}) \end{aligned}$$

We perform our calculus to check whether the concept $\mathit{Bears} \sqcap \mathit{EndangeredSpecies}$ is satisfiable with respect to $\mathrm{circ}_{\mathsf{CP}}(KB)$.

We start with the constraint system initialised as follows.

$$\begin{aligned} S_{KB} \cup \{x : \mathit{Bears} \sqcap \mathit{EndangeredSpecies}\} = \{\ & \mathit{BlueWhale} : \neg\mathit{Bears}\,, \\ & \mathit{BlueWhale} : \mathit{EndangeredSpecies}\,,\ x : \mathit{Bears} \sqcap \mathit{EndangeredSpecies}\ \} \end{aligned}$$

From the application of the $\longrightarrow_{\sqcap}$-rule and subsequently of the $\longrightarrow_{<_{\mathsf{CP}}}$-rule, the following two resulting completions are produced.

$$\begin{aligned} S_1 = \{\ & \mathit{BlueWhale} : \neg\mathit{Bears}\,,\ \mathit{BlueWhale} : \mathit{EndangeredSpecies}\,,\ \mathit{BlueWhale} : \mathit{Bears}\ \} \\ S_2 = \{\ & \mathit{BlueWhale} : \neg\mathit{Bears}\,,\ \mathit{BlueWhale} : \mathit{EndangeredSpecies}\,, \\ & \iota_0 : \mathit{Bears}\,,\ \iota_0 : \mathit{EndangeredSpecies}\ \} \end{aligned}$$

The completion S_1 obviously contains an inconsistency clash, since it contains both the constraints $\mathit{BlueWhale} : \mathit{Bears}$ and $\mathit{BlueWhale} : \neg\mathit{Bears}$.

For the completion S_2, we construct KB' according to Algorithm 1 as follows.

$$\begin{aligned} KB' = \{\ & \neg\mathit{Bears}(\mathit{BlueWhale})\,,\ \mathit{EndangeredSpecies}(\mathit{BlueWhale})\,, \\ & \mathit{EndangeredSpecies} \sqsubseteq \{\mathit{BlueWhale}\,, \iota_0\}\,, \\ & \neg\mathit{EndangeredSpecies} \sqcap \{\mathit{BlueWhale}\,, \iota_0\}(\iota) \qquad \} \end{aligned}$$

It can be verified by classical reasoning techniques that KB' has a model when the new individual ι is not uniquely named serving as a variable, and thus, the completion S_2 contains a preference clash.

Since both S_1 and S_2 contain some clash, the initial constraint system $S_{KB} \cup \{x : \mathit{Bears} \sqcap \mathit{EndangeredSpecies}\}$ has no clash-free completion. Hence, the concept $\mathit{Bears} \sqcap \mathit{EndangeredSpecies}$ is unsatisfiable with respect to $\mathrm{circ}_{\mathsf{CP}}(KB)$.

In the description logic literature, tableaux methods for sound and complete reasoning have been proposed for various DL variants including $\mathcal{ALCO}$. They detect inconsistencies in DL knowledge bases by checking completions of constraint systems for the occurrence of a clash. We include this result adapted to our setting in form of the following proposition.

Proposition 4 ($\mathcal{ALCO}$ correctness). *Let KB be an $\mathcal{ALCO}$ knowledge base and S be the completion of a constraint system containing at least the constraints of S_{KB}, with regard to any circumscription pattern and KB. Then S has a solution if and only if it contains neither an inconsistency clash nor an individual clash.*

Proof (Sketch). The top part of Table 1 (without the $\longrightarrow_{<_{\mathsf{CP}}}$-rule) captures the algorithm from [9], which is known to be correct. In fact, the proof from [9] essentially carries over.

Based on this correspondence between clash-free completions and their solutions, we can establish the correlation between solvability of constraint systems and the absence of preference clashes in their completions as the main result of this paper by the following proposition.

Proposition 5 (circumscriptive $\mathcal{ALCO}$ correctness). *Let KB be an $\mathcal{ALCO}$ knowledge base,* CP *be a circumscription pattern and S be the completion of a constraint system containing at least the constraints of S_{KB}, with regard to* CP *and KB. S is* CP*-solvable with respect to KB if and only if it contains no inconsistency clash, no individual clash and no preference clash with respect to* CP *and KB.*

Proof.
$\Rightarrow$: Assume that S is CP-solvable with respect to KB. According to Definition 4 there is a solution $(\mathcal{I}, \alpha^{\mathcal{I}})$ for S, such that $\mathcal{I}$ is a model of KB and there is no model $\mathcal{J}$ of KB with $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$. From Proposition 4, we know that S does neither contain an inconsistency clash nor an individual clash. We show by contradiction that S does also not contain a preference clash.

Assume that S contains a preference clash with respect to CP and KB. Then, $S_{KB'}[\iota/x]$ has a completion S' with regard to $\mathsf{CP} = (\emptyset, \emptyset, M \cup F \cup V)$ and KB' that contains no inconsistency and no individual clash, where the knowledge base KB' is constructed based on CP and KB according to Algorithm 1. Observe that, by construction, $KB \subset KB'$ and that ι is a new individual in KB' that cannot occur in KB. Hence, we have that $S_{KB} \subset S_{KB'}[\iota/x] \subseteq S'$. Proposition 4($\Leftarrow$) implies that there is a solution $(\mathcal{J}, \alpha^{\mathcal{J}})$ for S', since S' is clash-free. Due to Lemma 1, and since $S_{KB} \subseteq S'$, it follows that $\mathcal{J}$ is a model of both KB' and KB. It remains to show that $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$, to contradict the containment of a preference clash in S. Without loss of generality, we can assume that $\Delta^{\mathcal{I}} = \Delta^{\mathcal{J}}$ and that $a^{\mathcal{I}} = a^{\mathcal{J}}$ for all individuals $a \in \mathsf{N}_I$. Moreover, we assumed $F = \emptyset$ due to Proposition 1, such that $\bar{A}^{\mathcal{J}} = \bar{A}^{\mathcal{I}}$ for all $\bar{A} \in F$ vacuously holds. We prove the following claims: a) $\tilde{A}^{\mathcal{J}} \subseteq \tilde{A}^{\mathcal{I}}$ for all $\tilde{A} \in M$, and b) $\tilde{A}^{\mathcal{J}} \subset \tilde{A}^{\mathcal{I}}$ for some $\tilde{A} \in M$.

- a) Due to the inclusion axioms for minimised concepts inserted into KB' by Algorithm 1, and since $\mathcal{J}$ is a model of KB', $\mathcal{J}$ has the property $\tilde{A}^{\mathcal{J}} \subseteq \{\alpha^{\mathcal{J}}(a) \mid a : \tilde{A} \in S\}$ for each $\tilde{A} \in M_{KB}$. For every $\tilde{A} \in M_{KB}$, all the constraints $a : \tilde{A} \in S$ are satisfied by $(\mathcal{I}, \alpha^{\mathcal{I}})$, i.e. $\alpha^{\mathcal{I}}(a) \in \tilde{A}^{\mathcal{I}}$, and therefore we have that $\{\alpha^{\mathcal{I}}(a) \mid a : \tilde{A} \in S\} \subseteq \tilde{A}^{\mathcal{I}}$. Since $\alpha^{\mathcal{I}}$ and $\alpha^{\mathcal{J}}$ coincide on individuals, it follows that $\tilde{A}^{\mathcal{J}} \subseteq \tilde{A}^{\mathcal{I}}$ for all $\tilde{A} \in M_{KB}$.
- b) By construction of KB', $S_{KB'}[\iota/x]$ contains a constraint $x : \bigsqcup_{\tilde{A}} D_{\tilde{A}}$, and for one of the disjuncts $D_{\tilde{A}}$ its completion S' contains a constraint of the form

$x : \{a_1, \ldots, a_n\} \sqcap \neg\tilde{A}$ with $a_i : \tilde{A} \in S$ for $i = 1 \ldots n$. Since S' is a completion to which none of the tableaux rules apply, the $\longrightarrow_{\sqcap}$- and the $\longrightarrow_{\mathcal{O}}$-rule have produced the constraints $a : \{a_1, \ldots, a_n\}$ and $a : \neg\tilde{A}$ in S' in which the variable x has been replaced by an individual a. As a solution for S', $(\mathcal{J}, \alpha^{\mathcal{J}})$ satisfies these two constraints and we have that both $\alpha^{\mathcal{J}}(a) \in (\Delta^{\mathcal{J}} \setminus \tilde{A}^{\mathcal{J}})$ and $\alpha^{\mathcal{J}}(a) \in \{\alpha^{\mathcal{J}}(a) \mid a : \tilde{A} \in S\}$ hold. This implies that $\alpha^{\mathcal{J}}(a) \notin \tilde{A}^{\mathcal{J}}$ and, since $(\mathcal{I}, \alpha^{\mathcal{I}})$ satisfies the constraint $a : \tilde{A}$, that $\alpha^{\mathcal{J}}(x) = \alpha^{\mathcal{I}}(a) \in \tilde{A}^{\mathcal{I}}$. From the arguments under b) we already know that $\tilde{A}^{\mathcal{J}} \subseteq \tilde{A}^{\mathcal{I}}$, and since we have an element $a^{\mathcal{I}}$ which is in $\tilde{A}^{\mathcal{I}}$ but not in $\tilde{A}^{\mathcal{J}}$, it follows that $\tilde{A}^{\mathcal{J}} \subset \tilde{A}^{\mathcal{I}}$.

$\Leftarrow$: Let S contain no clash. From Proposition 4 we know that there is a solution $(\mathcal{I}, \alpha^{\mathcal{I}})$ for S. We show by contradiction that there is no model $\mathcal{J}$ of *KB* such that $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$.

Assume that there is a model $\mathcal{J}$ of *KB* with $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$. First we show that for some $\mathcal{J}$-assignment $\alpha^{\mathcal{J}}$, $(\mathcal{J}, \alpha^{\mathcal{J}})$ is a solution for $S_{KB'}[\iota/x]$, where the knowledge base KB' is constructed according to Algorithm 1. Due to $\mathcal{J} <_{\mathsf{CP}} \mathcal{I}$ we know that $\Delta^{\mathcal{J}} = \Delta^{\mathcal{I}}$ and $a^{\mathcal{J}} = a^{\mathcal{I}}$ for all individuals $a \in \Delta^{\mathcal{I}}$, and that for some $\tilde{A} \in M$ there is an element $\iota^{\mathcal{J}} \in \Delta^{\mathcal{J}}$ which is in $\tilde{A}^{\mathcal{I}}$ but not in $\tilde{A}^{\mathcal{J}}$. Let $\alpha^{\mathcal{J}}_{x,\iota}$ be a $\mathcal{J}$-assignment with $\alpha^{\mathcal{J}}_{x,\iota}(x) = \iota^{\mathcal{J}}$. Since $\mathcal{J}$ is a model of *KB*, $(\mathcal{J}, \alpha^{\mathcal{J}}_{x,\iota})$ is a solution for S_{KB} due to Lemma 1. Moreover, as the individual ι is new to KB' and $KB \subset KB'$ by construction of KB', the replacement of ι by x does not affect any constraint in S_{KB} and we have that $S_{KB} \subset S_{KB'}[\iota/x]$. Hence, it suffices to show that the constraints in $S_{KB'}[\iota/x] \setminus S_{KB}$ are satisfied by $(\mathcal{J}, \alpha^{\mathcal{J}}_{x,\iota})$. For this purpose, we consider the axioms in $KB' \setminus KB$ that are inserted into KB' by Algorithm 1, and that can be a) concept inclusion axioms of the form $\tilde{A} \sqsubseteq \{a_1, \ldots, a_n\}$, or b) the concept assertion axiom $(\bigsqcup_{\tilde{A}} D_{\tilde{A}})(\iota)$ with disjuncts $D_{\tilde{A}}$ of the form $\neg\tilde{A} \sqcap \{a_1, \ldots, a_n\}$, for individuals $\{a_i \mid a_i : \tilde{A} \in S\}$ with $i \in \{1, \ldots, n\}$.

- a) For every $\tilde{A} \in M_{KB}$, KB' contains an axiom $\tilde{A} \sqsubseteq \{a_1, \ldots, a_n\}$ with individuals a_i that occur in concept constraints of the form $a_i : \tilde{A}$ within S. Since S is a completion, in any constraint of the form $x : \tilde{A}$ the variable x has been replaced by an individual $a \in \mathsf{N}_I^S$ in S due to the $\longrightarrow_{<_{\mathsf{CP}}}$-rule, such that for any constraint $o : \tilde{A} \in S$ we have that $o = a_i$ for some $i \in \{1, \ldots, n\}$. Since $\mathcal{I}$ is a solution for S, we have that $\tilde{A}^{\mathcal{I}} \subseteq \{\alpha^{\mathcal{I}}(a_1), \ldots, \alpha^{\mathcal{I}}(a_n)\} = \{a_1^{\mathcal{I}}, \ldots, a_n^{\mathcal{I}}\}$. Since $\tilde{A}^{\mathcal{J}} \subseteq \tilde{A}^{\mathcal{I}}$ holds by assumption, $\mathcal{J}$ satisfies $\tilde{A}^{\mathcal{J}} \subseteq \{a_1^{\mathcal{I}}, \ldots, a_n^{\mathcal{I}}\}$, and thus, the axiom $\tilde{A} \sqsubseteq \{a_1, \ldots, a_n\}$ for every $\tilde{A} \in M_{KB}$. If there are no assertions $\tilde{A}(a_i)$ in KB' then $\tilde{A}^{\mathcal{I}} = \emptyset$ and the respective axiom has the form $\tilde{A} \sqsubseteq \bot$. Hence, $(\mathcal{J}, \alpha^{\mathcal{J}}_{x,\iota})$ satisfies all the constraints $\forall x.x : C$ that result from these inclusion axioms in $S_{KB'}[\iota/x]$.
- b) Furthermore, due to the concept assertion $(\bigsqcup_{\tilde{A}} D_{\tilde{A}})(\iota)$ in KB', $S_{KB'}[\iota/x]$ contains the constraint $x : \bigsqcup_{\tilde{A}} D_{\tilde{A}}$ with disjuncts $D_{\tilde{A}}$ of the form $\neg\tilde{A} \sqcap \{a_1, \ldots, a_n\}$. Since from b) we know that $\tilde{A}^{\mathcal{I}} \subseteq \{a_1^{\mathcal{I}}, \ldots, a_n^{\mathcal{I}}\}$, and since $a^{\mathcal{I}} = a^{\mathcal{J}}$ for all individuals a, we get that $\tilde{A}^{\mathcal{I}} = \{a_1^{\mathcal{J}}, \ldots, a_n^{\mathcal{J}}\} \subseteq \Delta^{\mathcal{J}}$. As for some $\tilde{A} \in M_{KB}$ the element $\iota^{\mathcal{J}}$ is in $\tilde{A}^{\mathcal{I}}$ but not in $\tilde{A}^{\mathcal{J}}$, we have that $\alpha^{\mathcal{J}}_{x,\iota}(x) \in \tilde{A}^{\mathcal{I}} \setminus \tilde{A}^{\mathcal{J}}$, and thus, $\alpha^{\mathcal{J}}_{x,\iota}(x) \in (\{a_1^{\mathcal{J}}, \ldots, a_k^{\mathcal{J}}\} \setminus \tilde{A}^{\mathcal{J}}) = (\Delta^{\mathcal{J}} \setminus \tilde{A}^{\mathcal{J}}) \cap$

$\{\alpha^{\mathcal{J}}_{x,\iota}(a_1), \dots, \alpha^{\mathcal{J}}_{x,\iota}(a_k)\}$. Hence, the pair $(\mathcal{J}, \alpha^{\mathcal{J}}_{x,\iota})$ satisfies the constraint $x : \bigsqcup_{\tilde{A}} D_{\tilde{A}}$ for some $\tilde{A} \in M$ with $\tilde{A}^{\mathcal{J}} \subset \tilde{A}^{\mathcal{I}}$, as one of its disjuncts is satisfied.

Having shown that $(\mathcal{J}, \alpha^{\mathcal{J}}_{x,\iota})$ is a solution for $S_{KB'}[\iota/x]$, from Proposition 3($\Rightarrow$) and from Proposition 4($\Rightarrow$) it follows that there is a clash-free completion of $S_{KB'}[\iota/x]$. Hence, S must contain a preference clash, which contradicts the existence of $\mathcal{J}$. $\square$

As a direct result of the propositions 2, 3, 5 and Lemma 2, we obtain that the presented calculus provides an effective procedure for reasoning with circumscribed knowledge bases, reflected by the following theorem.

Theorem 1 (soundness/completeness). *Let KB be an $\mathcal{ALCO}$ knowledge base, CP be a circumscription pattern and C be an $\mathcal{ALCO}$ concept. C is satisfiable with respect to $circ_{\mathsf{CP}}(KB)$ if and only if the algorithm based on Table 1 results in a clash-free completion of the constraint system $S_{KB} \cup \{x : C\}$.*

By Theorem 1, the proposed tableaux calculus is a decision procedure for reasoning in $\mathcal{ALCO}$ with concept circumscription.

4 Conclusion

We have presented a tableaux calculus for concept satisfiability with respect to circumscribed DL knowledge bases in the logic $\mathcal{ALCO}$. Building on tableaux procedures for classical DLs, the calculus checks a constraint system not only for clashes due to inconsistent concept assertion and individual naming, but also for preference clashes, which occur whenever the model associated with the constraint system is not minimal with respect to the preference relation $<_{\mathsf{CP}}$. This check is performed by testing a specifically constructed classical $\mathcal{ALCO}$ knowledge base for satisfiability, which requires reasoning in classical DL with nominals and equality between individuals.

We have proved that the presented calculus is sound and complete for verifying concept satisfiability in circumscriptive $\mathcal{ALCO}$. By this we have devised a first guided algorithmisation for description logic with circumscription that integrates well with state of the art tableaux methods for DL reasoning. This lays a basis for further investigations on optimisation of the calculus within the framework of tableaux procedures as a guided way for model construction. We have implemented a first prototype[6] of the calculus in Java that works together with ontology development tools, such as Protégé, via the DIG interface.

As future work we see the update of the calculus to support more expressive features, such as prioritisation between minimised concepts or the remaining constructs of the Web Ontology Language OWL [16]. Moreover, optimisation issues need to be addressed to obtain a more efficient reasoning procedure. First ideas for specific optimisations would be to employ model caching techniques for the inner classical tableaux step as KB' might be identical in multiple cases, to

[6] Available at `http://www.fzi.de/downloads/wim/sgr/CircDL.zip` .

postpone assertions of individuals to minimised predicates in order to avoid constructing non-minimal models, and to exploit early closing of tableaux branches through preference clash detection. Besides these, it would be interesting to see how well preferential tableaux performs when included in optimised state-of-the-art DL reasoners. Furthermore, a methodology remains to be developed for the formulation of appropriate circumscription patterns in various cases of nonmonotonic reasoning, as pointed out in [2].

References

1. F. Baader and B. Hollunder. Embedding Defaults into Terminological Knowledge Representation Formalisms. *Journal of Automated Reasoning*, 14(1):149–180, 1995.
2. P. Bonatti, C. Lutz, and F. Wolter. Expressive Non-Monotonic Description Logics Based on Circumscription. In *Proc. of the Tenth Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'06)*, pages 400–410, 2006.
3. J. de Kleer and K. Konolige. Eliminating the Fixed Predicates from a Circumscription. *Artif. Intell.*, 39(3):391–398, 1989.
4. F. M. Donini, M. Lenzerini, D. Nardi, W. Nutt, and A. Schaerf. An Epistemic Operator for Description Logics. *Artificial Intelligence*, 100(1-2):225–274, 1998.
5. F. M. Donini, D. Nardi, and R. Rosati. Description Logics of Minimal Knowledge and Negation as Failure. *ACM Trans. on Computational Logic*, 3(2):177–225, 2002.
6. S. Grimm. *Semantic Matchmaking with Nonmonotonic Description Logics.* IOS Press, 2007.
7. S. Grimm and P. Hitzler. Semantic Matchmaking of Resources with Local Closed-World Reasoning. *Int. Journal of eCommerce (IJEC)*, 12(2):89–126, 2008.
8. P. Hitzler, M. Krötzsch, and S. Rudolph. *Foundations of Semantic Web Technologies.* Chapman & Hall/CRC, 2009.
9. I. Horrocks and U. Sattler. Ontology reasoning in the SHOQ(D) description logic. In *Proc. IJCAI-01*, pages 199–204, 2001.
10. M. Knorr, J. J. Alferes, and P. Hitzler. A coherent well-founded model for hybrid MKNF knowledge bases. In M. Ghallab et al., editors, *Proceedings of the 18th European Conference on Artificial Intelligence, ECAI2008*, pages 99–103. IOS Press, 2008.
11. J. McCarthy. Circumscription – A Form of Non-Monotonic Reasoning. *Artificial Intelligence*, 13(1–2):27–39, 1980.
12. B. Motik and R. Rosati. A faithful integration of description logics with logic programming. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07)*, pages 477–482, Hyderabad, India, January 6–12 2007. AAAI Press.
13. I. Niemelä. Implementing Circumscription Using a Tableau Method. In *Proc. of the 12th Europ. Conf. on Artificial Intelligence (ECAI'96).* J. Wiley & Sons, 1996.
14. A. Schaerf. Reasoning with Individuals in Concept Languages. *Data Knowl. Eng.*, 13(2):141–176, 1994.
15. M. Schmidt-Schauß; and G. Smolka. Attributive Concept Descriptions with Complements. *Artif. Intell.*, 48(1):1–26, 1991.
16. W3C OWL Working Group. OWL 2 Web Ontology Language: Document Overview. `http://www.w3.org/TR/owl2-overview/`, 2009.