Fachbereich Informatik der Universität Hamburg

Vogt-Kölln-Str. 30 ◊ D-22527 Hamburg / Germany

University of Hamburg - Computer Science Department

Mitteilung Nr. 290/2000 • Memo No. 290/2000

The Description Logic $\mathcal{ALCNH}_{R^+}$ Extended with Concrete Domains

Volker Haarslev, Ralf Möller and Michael Wessel

Arbeitsbereich KOGS

FBI-HH-M-290/00

June 2000

The Description Logic $\mathcal{ALCNH}_{R^+}$ Extended with Concrete Domains

Volker Haarslev, **Ralf Möller** and **Michael Wessel**
University of Hamburg, Computer Science Department,
Vogt-Kölln-Str. 30, 22527 Hamburg, Germany

Abstract

The paper introduces the description logic $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$. Prominent language features beyond $\mathcal{ALC}$ are number restrictions, role hierarchies, transitively closed roles, generalized concept inclusions and concrete domains. As in other languages based on concrete domains (e.g. $\mathcal{ALC}(\mathcal{D})$) a so-called predicate exists restriction concept constructor is provided. However, compared to $\mathcal{ALC}(\mathcal{D})$ only features and no feature chains are allowed in this operator. This results in a limited expressivity w.r.t. concrete domains but is required to ensure the decidability of the language. We show that the results can be exploited for building practical description logic systems for solving e.g. configuration problems.

1 Introduction

In the field of knowledge representation, description logics (DLs) have been proven to be a sound basis for solving application problems. Detailed introductions to description logics can be found in [Woods and Schmolze, 1992] and [Donini et al., 1996]. An application domain where DLs have been successfully applied is *configuration* (see [Wright et al., 1993] for an early publication). In the following we assume the reader is familiar with description logics (see also [Baader, 1999; Baader and Sattler, 2000] for recent introductions). The main notions for domain modeling are concepts (unary predicates) and roles (binary predicates). Furthermore, a set of axioms (also called TBox) is used for modeling the terminology of an application. Knowledge about specific individuals and their interrelationships is modeled with a set of additional axioms (so-called ABox).

Experiences with description logics in applications indicate that negation, existential and universal restrictions, transitive roles, role hierarchies, and number restrictions are required to solve practical modeling problems without resorting to ad hoc extensions. The description logics $\mathcal{ALCNR}$ [Buchheit et al., 1993] and $\mathcal{ALCNH}_{R^+}$ [Haarslev and Möller, 2000a] formalize many of the above-mentioned requirements. The DL knowledge representation system RACE provides an optimized implementation for ABox reasoning in $\mathcal{ALCNH}_{R^+}$ [Haarslev et al., 1999; Haarslev and Möller, 2000a]. RACE can be used for large-scale knowledge modeling [Haarslev and Möller, 2000b]. A calculus for ABox reasoning for the logic $\mathcal{SHIQ}$ (i.e. $\mathcal{ALCNH}_{R^+}$ augmented with qualified number restrictions and inverse roles) has been introduced in [Horrocks et al., 2000]. However, an implementation of the $\mathcal{SHIQ}$ ABox reasoning algorithm is not yet available.

The requirements derived from practical applications of DLs ask for even more expressive languages. It is well-known that reasoning about objects from other domains (so-called concrete domains, e.g. for the reals) is very important for practical applications as well [Baader and Hanschke, 1991a; Baader and Hanschke, 1991b]. Thus, an extension of the $\mathcal{ALCNH}_{R^+}$ knowledge representation system RACE with concrete domain is investigated.

Unfortunately, adding concrete domains to expressive description logics might lead to undecidable inference problems. For instance, in [Baader and Hanschke, 1992] it is proven that the logic $\mathcal{ALC}(\mathcal{D})$ plus an operator for the transitive closure of roles is undecidable. $\mathcal{ALCNH}_{R^+}$ offers transitive roles but no operator for the transitive closure of roles (see [Sattler, 1996, p. 342] for a detailed discussion about expressivity differences). In [Lutz, 1999] it is shown that $\mathcal{ALC}(\mathcal{D})$ with generalized inclusion axioms (GCIs) is undecidable. Thus, if termination and soundness are to be retained, there is no way extending an $\mathcal{ALCNH}_{R^+}$ DL system such as RACE with concrete domains as in $\mathcal{ALC}(\mathcal{D})$ without losing completeness. Even if GCIs were discarded, $\mathcal{ALCNH}_{R^+}$ with concrete domains would be undecidable because $\mathcal{ALCNH}_{R^+}$ offers role hierarchies and transitive roles, which provide the same expressivity as GCIs. With role hierarchies it is possible to (implicitly) declare a universal role, which can be used in combination with a value restriction to achieve the same effect as with GCIs.

Thus, $\mathcal{ALCNH}_{R^+}$ can only be extended with concrete domain operators with limited expressivity. In order to support practical modeling requirements at least to some extent, we pursue a pragmatic approach by supporting a limited kind of concept exists restriction which supports only features (and no feature chains as in $\mathcal{ALC}(\mathcal{D})$, for details see below). The resulting language is called $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$. By proving soundness and completeness (and termination) of a tableaux calculus, the decidability of inference problems w.r.t. the language $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ is proved. As shown in this report, $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ can be used, for instance, as a basis for building practical application systems for solving certain classes of configuration problems, see also [Buchheit et al., 1995; Schröder et al., 1996].

2 The Description Logic $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$

The description logic $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ augments the basic logic $\mathcal{ALC}$ [Schmidt-Schauss and Smolka, 1991] with number restrictions, role hierarchies, transitively closed roles and concrete domains. The use of number restrictions in combination with transitive roles and role hierarchies is syntactically restricted: no number restrictions are possible for (i) transitive roles and (ii) for any role which has a transitive subrole (see also [Horrocks et al., 1999]). In addition to the operators known from $\mathcal{ALCNH}_{R^+}$ a limited kind of predicate exists restriction operator for concrete domains is supported. Furthermore, we assume that the unique name assumption holds for the individuals explicitly mentioned in an ABox.

2.1 The Concept Language

For presenting the syntax and semantics of the language $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ a few definitions are required.

Definition 1 (Features, Roles, Role Axioms, Role Hierarchy) Let F and R be disjoint sets of *feature names* and *role names*, respectively. For brevity, a role name is also called a role and a feature name is also called a feature. Furthermore, let $S \subseteq R$ be the set of *simple roles*. If R and S are role names, then $\mathsf{R} \sqsubseteq \mathsf{S}$ is a *role inclusion axiom*. If R is a role name, then $\mathsf{transitive}(\mathsf{R})$ is called a *role transitivity axiom*. Both kinds of axioms are called *role axioms*. A set of role inclusion axioms is also called a *role hierarchy*.

Additionally, we define the set of ancestors and descendants of a role as well as the set of transitive roles w.r.t. a set of role axioms.

Definition 2 (Role Descendants/Ancestors) Let $\mathcal{R}$ be a set of role axioms and $\sqsubseteq_\mathcal{R}$ be defined as $\{(\mathsf{R},\mathsf{S}) \mid \mathsf{R} \sqsubseteq \mathsf{S} \in \mathcal{R}\}$ and let $\sqsubseteq^*{}_\mathcal{R}$ be the reflexive transitive closure of $\sqsubseteq_\mathcal{R}$ over R. Given a set of role axioms $\mathcal{R}$ the set $\mathsf{R}^\uparrow{}_\mathcal{R} := \{\mathsf{S} \in R \mid (\mathsf{R},\mathsf{S}) \in \sqsubseteq^*{}_\mathcal{R}\}$ defines the *ancestors* and $\mathsf{R}^\downarrow{}_\mathcal{R} := \{\mathsf{S} \in R \mid (\mathsf{S},\mathsf{R}) \in \sqsubseteq^*{}_\mathcal{R}\}$ the *descendants* of a role R w.r.t. a set of role axioms $\mathcal{R}$. The set of transitive roles $T_\mathcal{R}$ of a set of role axioms $\mathcal{R}$ is defined as $\{\mathsf{R} \mid \mathsf{transitive}(\mathsf{R}) \in \mathcal{R}\}$.

Definition 3 (Role Box) A finite set of role axioms is called a *role box* $\mathcal{R}$ if $\forall\, \mathsf{R} \in S : \mathsf{R}^\downarrow{}_\mathcal{R} \cap T = \emptyset$. A role box is also called *RBox* for brevity.

In the following, the index $_\mathcal{R}$ is omitted if the role box $\mathcal{R}$ is clear from the context.

In accordance with [Baader and Hanschke, 1991a] we also define the notion of a concrete domain.

Definition 4 (Concrete Domain) A *concrete domain* $\mathcal{D}$ is a pair $(\Delta_\mathcal{D}, \Phi_\mathcal{D})$, where $\Delta_\mathcal{D}$ is a set called the domain, and $\Phi_\mathcal{D}$ is a set of predicate names. Each predicate name $\mathsf{P}_\mathcal{D}$ from $\Phi_\mathcal{D}$ is associated with an arity n and an n-ary predicate $\mathsf{P}_\mathcal{D}$. A concrete domain $\mathcal{D}$ is called *admissible* iff:

- The set of predicate names $\Phi_\mathcal{D}$ is closed under negation and $\Phi_\mathcal{D}$ contains a name $\top_\mathcal{D}$ for $\Delta_\mathcal{D}$,
- The satisfiability problem $\mathsf{P}_1^{\mathsf{n}_1}(\mathsf{x}_{11}, \ldots, \mathsf{x}_{1\mathsf{n}_1}) \wedge \ldots \wedge \mathsf{P}_\mathsf{m}^{\mathsf{n}_\mathsf{m}}(\mathsf{x}_{\mathsf{m}1}, \ldots, \mathsf{x}_{\mathsf{m}\mathsf{n}_\mathsf{m}})$ is decidable (m is finite, $\mathsf{P}_\mathsf{i}^{\mathsf{n}_\mathsf{i}} \in \Phi_\mathcal{D}$, n_i is the arity of P, and x_{jk} is a name for an object from $\Delta_\mathcal{D}$).

We assume that $\bot_\mathcal{D}$ is the negation of the predicate $\top_\mathcal{D}$. Using the definitions from above, the syntax of concept terms in $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ is defined as follows.

Definition 5 (Concept Terms) Let C be a set of concept names which is disjoint from R and F. Any element of C is a *concept term*. If C and D are concept terms, $\mathsf{R} \in R$ is an arbitrary role, $\mathsf{S} \in S$ is a simple role, $n, m \in \mathbb{N}, n > 1$, and $m > 0$, $\mathsf{P} \in \Phi_\mathcal{D}$ is a predicate of the concrete domain, $\mathsf{f}, \mathsf{f}_1, \ldots, \mathsf{f}_\mathsf{k} \in F$ are features, then the following expressions are also concept terms:

- $\mathsf{C} \sqcap \mathsf{D}$ *(conjunction)*
- $\mathsf{C} \sqcup \mathsf{D}$ *(disjunction)*
- $\neg \mathsf{C}$ *(negation)*
- $\forall \mathsf{R} \, . \, \mathsf{C}$ *(concept value restriction)*
- $\exists \mathsf{R} \, . \, \mathsf{C}$ *(concept exists restriction)*
- $\exists_{\leq m} \mathsf{S}$ *(at most number restriction)*
- $\exists_{\geq n} \mathsf{S}$ *(at least number restriction).*
- $\exists \mathsf{f}_1, \ldots, \mathsf{f}_\mathsf{k} \, . \, \mathsf{P}$ *(predicate exists restriction).*
- $\forall \mathsf{f} \, . \, \bot_\mathcal{D}$ *(no concrete domain filler restriction).*

A concept term may be put in parentheses. For brevity, concept terms are also called concepts. $\top$ ($\bot$) is considered as an abbreviation for $\mathsf{C} \sqcup \neg \mathsf{C}$ ($\mathsf{C} \sqcap \neg \mathsf{C}$) for some $\mathsf{C} \in C$. For an arbitrary role R, the term $\exists_{\geq 1} \mathsf{R}$ can be rewritten as $\exists \mathsf{R} \, . \, \top$, $\exists_{\geq 0} \mathsf{R}$ as $\top$, and $\exists_{\leq 0} \mathsf{R}$ as $\forall \mathsf{R} \, . \, \bot$. Thus, we do not consider these terms as number restrictions in our language.

Definition 6 (Terminological Axiom, TBox) If C and D are concept terms, then $\mathsf{C} \sqsubseteq \mathsf{D}$ is a terminological axiom. A terminological axiom is also called *generalized concept inclusion* or *GCI*. A finite set of terminological axioms $\mathcal{T}$ is called a *terminology* or *TBox*.

The next definition gives a model-theoretic semantics to the language introduced above. Let $\mathcal{D} = (\Delta_{\mathcal{D}}, \Phi_{\mathcal{D}})$ be a concrete domain.

Definition 7 (Semantics) An *interpretation* $\mathcal{I}_{\mathcal{D}} = (\Delta_{\mathcal{I}}, \Delta_{\mathcal{D}}, \cdot^{\mathcal{I}})$ consists of a set $\Delta_{\mathcal{I}}$ (the abstract domain), a set $\Delta_{\mathcal{D}}$ (the domain of the 'concrete domain' $\mathcal{D}$) and an interpretation function $\cdot^{\mathcal{I}}$. The interpretation function $\cdot^{\mathcal{I}}$maps each concept name C to a subset $\mathsf{C}^{\mathcal{I}}$ of $\Delta_{\mathcal{I}}$, each role name R from R to a subset $\mathsf{R}^{\mathcal{I}}$ of $\Delta_{\mathcal{I}} \times \Delta_{\mathcal{I}}$. Each feature f from F is mapped to a partial function $\mathsf{f}^{\mathcal{I}}$ from $\Delta_{\mathcal{I}}$ to $\Delta_{\mathcal{D}}$ where $\mathsf{f}^{\mathcal{I}}(\mathsf{a}) = \mathsf{x}$ will be written as $(\mathsf{a}, \mathsf{x}) \in \mathsf{f}^{\mathcal{I}}$. Each predicate name P from $\Phi_{\mathcal{D}}$ with arity n is mapped to a subset $\mathsf{P}^{\mathcal{I}}$ of $\Delta_{\mathcal{D}}^n$. Let the symbols C, D be concept expressions, R, S be role names, $\mathsf{f}, \mathsf{f_1}, \ldots, \mathsf{f_n}$ be features and let P be a predicate name. Then, the interpretation function is extended to arbitrary concept and role terms as follows ($\|\cdot\|$ denotes the cardinality of a set):

$$
\begin{aligned}
(\mathsf{C} \sqcap \mathsf{D})^{\mathcal{I}} &:= \mathsf{C}^{\mathcal{I}} \cap \mathsf{D}^{\mathcal{I}} \\
(\mathsf{C} \sqcup \mathsf{D})^{\mathcal{I}} &:= \mathsf{C}^{\mathcal{I}} \cup \mathsf{D}^{\mathcal{I}} \\
(\neg \mathsf{C})^{\mathcal{I}} &:= \Delta_{\mathcal{I}} \setminus \mathsf{C}^{\mathcal{I}} \\
(\exists\, \mathsf{R} \,.\, \mathsf{C})^{\mathcal{I}} &:= \{a \in \Delta_{\mathcal{I}} \,|\, \exists\, b : (a, b) \in \mathsf{R}^{\mathcal{I}}, b \in \mathsf{C}^{\mathcal{I}}\} \\
(\forall\, \mathsf{R} \,.\, \mathsf{C})^{\mathcal{I}} &:= \{a \in \Delta_{\mathcal{I}} \,|\, \forall\, b : (a, b) \in \mathsf{R}^{\mathcal{I}} \Rightarrow b \in \mathsf{C}^{\mathcal{I}}\} \\
(\exists_{\geq n}\, \mathsf{R})^{\mathcal{I}} &:= \{a \in \Delta_{\mathcal{I}} \,|\, \|\{b \,|\, (a, b) \in \mathsf{R}^{\mathcal{I}}\}\| \geq n\} \\
(\exists_{\leq m}\, \mathsf{R})^{\mathcal{I}} &:= \{a \in \Delta_{\mathcal{I}} \,|\, \|\{b \,|\, (a, b) \in \mathsf{R}^{\mathcal{I}}\}\| \leq m\} \\
(\exists\, \mathsf{f_1}, \ldots, \mathsf{f_n} \,.\, \mathsf{P})^{\mathcal{I}} &:= \{a \in \Delta_{\mathcal{I}} \,|\, \exists x_1, \ldots, x_n \in \Delta_{\mathcal{D}} : \\
&\qquad\qquad (a, x_1) \in \mathsf{f_1}^{\mathcal{I}}, \ldots, (a, x_n) \in \mathsf{f_n}^{\mathcal{I}}, \\
&\qquad\qquad (x_1, \ldots, x_n) \in \mathsf{P}^{\mathcal{I}}\} \\
(\forall\, \mathsf{f} \,.\, \bot_{\mathcal{D}})^{\mathcal{I}} &:= \{a \in \Delta_{\mathcal{I}} \,|\, \neg \exists x_1 \in \Delta_{\mathcal{D}} : (a, x_1) \in \mathsf{f}^{\mathcal{I}}\}
\end{aligned}
$$

An interpretation $\mathcal{I}_{\mathcal{D}}$ is a *model of a concept* C iff $\mathsf{C}^{\mathcal{I}_{\mathcal{D}}} \neq \emptyset$. An interpretation $\mathcal{I}_{\mathcal{D}}$ satisfies a terminological axiom $\mathsf{C} \sqsubseteq \mathsf{D}$ iff $\mathsf{C}^{\mathcal{I}} \subseteq \mathsf{D}^{\mathcal{I}}$. $\mathcal{I}_{\mathcal{D}}$ is a *model of a TBox* $\mathcal{T}$ iff it satisfies all terminological axioms $\mathsf{C} \sqsubseteq \mathsf{D}$ in $\mathcal{T}$. An interpretation $\mathcal{I}_{\mathcal{D}}$ is a *model of an RBox* $\mathcal{R}$ iff $\mathsf{R}^{\mathcal{I}} \subseteq \mathsf{S}^{\mathcal{I}}$ for all role inclusions $\mathsf{R} \sqsubseteq \mathsf{S}$ in $\mathcal{R}$ and, in addition, $\forall\, \mathsf{transitive}(\mathsf{R}) \in \mathcal{R} : \mathsf{R}^{\mathcal{I}} = (\mathsf{R}^{\mathcal{I}})^{+}$.

2.2 The Assertional Language

In the following, the language for representing knowledge about specific individuals is introduced.

Definition 8 (Assertional Axioms, ABox) Let $O = O_O \cup O_N$ be a set of individual names (or individuals), where the set O_O of "old" individuals is disjoint with the set O_N of "new" individuals. Old individuals are those names that explicitly appear in an ABox given as input to an algorithm for solving an inference problem (see below), i.e. the initially mentioned individuals must not be in O_N. Elements of O_N will be generated internally. Furthermore, let O_C be a set of names for concrete objects ($O_C \cap O = \emptyset$). If C is a concept term, $\mathsf{R} \in R$ a role name, $\mathsf{f} \in F$ a feature, $\mathsf{a}, \mathsf{b} \in O_O$ are individual names and $\mathsf{x}, \mathsf{x}_1, \ldots, \mathsf{x}_\mathsf{n} \in O_C$ are names for concrete objects, then the following expressions are *assertional axioms* or *ABox assertions*:

- $\mathsf{a} : \mathsf{C}$ *(concept assertion)*,
- $(\mathsf{a}, \mathsf{b}) : \mathsf{R}$ *(role assertion)*,
- $(\mathsf{a}, \mathsf{x}) : \mathsf{f}$ *(concrete domain feature assertion)*,
- $(\mathsf{x}_1, \ldots, \mathsf{x}_\mathsf{n}) : \mathsf{P}$ *(concrete domain predicate assertion)*.

An *ABox* $\mathcal{A}$ is a finite set of assertional axioms.

We need a few additional terms: An individual b is called a *direct successor* of an individual a in an ABox $\mathcal{A}$ iff $\mathcal{A}$ contains the assertional axiom $(\mathsf{a}, \mathsf{b}) : \mathsf{R}$. An individual b is called a *successor* of a if it is either a direct successor of a or there exists in $\mathcal{A}$ a chain of assertions $(\mathsf{a}, \mathsf{b}_1) : \mathsf{R}_1, (\mathsf{b}_1, \mathsf{b}_2) : \mathsf{R}_2, \ldots, (\mathsf{b}_\mathsf{n}, \mathsf{b}) : \mathsf{R}_{\mathsf{n}+1}$. In case that $\mathsf{R}_\mathsf{i} = \mathsf{R}_\mathsf{j}$ or $\mathsf{R}_\mathsf{i} \in \mathsf{R}^{\downarrow}$ for all $\mathsf{i}, \mathsf{j} \in 1..\mathsf{n}+1$ we call b the (direct) R-*successor* of a. A (direct) *predecessor* is defined analogously. Note that concrete domain objects are not considered as successors or predecessors in the sense defined above.

The interpretation function $\cdot^{\mathcal{I}}$ of the interpretation $\mathcal{I}_{\mathcal{D}}$ can be extended to the assertional language. Every individual name from O to a single element $\Delta_{\mathcal{I}}$ in such a way that for $\mathsf{a}, \mathsf{b} \in O_O$, $\mathsf{a}^{\mathcal{I}} \neq \mathsf{b}^{\mathcal{I}}$ if $\mathsf{a} \neq \mathsf{b}$ *(unique name assumption)*. This ensures that different individuals in O_O are interpreted as different objects. The unique name assumption does not hold for elements of O_N, i.e. for $\mathsf{a}, \mathsf{b} \in O_N$, $\mathsf{a}^{\mathcal{I}} = \mathsf{b}^{\mathcal{I}}$ may hold even if $\mathsf{a} \neq \mathsf{b}$. Concrete objects from O_C are mapped to elements of $\Delta_{\mathcal{D}}$.

An interpretation satisfies an assertional axiom $\mathsf{a} : \mathsf{C}$ iff $\mathsf{a}^{\mathcal{I}} \in \mathsf{C}^{\mathcal{I}}$, $(\mathsf{a}, \mathsf{b}) : \mathsf{R}$ iff $(\mathsf{a}^{\mathcal{I}}, \mathsf{b}^{\mathcal{I}}) \in \mathsf{R}^{\mathcal{I}}$, $(\mathsf{a}, \mathsf{x}) : \mathsf{f}$ iff $(\mathsf{a}^{\mathcal{I}}, \mathsf{x}^{\mathcal{I}}) \in \mathsf{f}^{\mathcal{I}}$ and $(\mathsf{x}_1, \ldots, \mathsf{x}_\mathsf{n}) : \mathsf{P}$ iff $({\mathsf{x}_1}^{\mathcal{I}}, \ldots, {\mathsf{x}_\mathsf{n}}^{\mathcal{I}}) \in \mathsf{P}^{\mathcal{I}}$. An interpretation $\mathcal{I}_{\mathcal{D}}$ is a *model of an ABox* $\mathcal{A}$ iff it satisfies all assertional axioms in $\mathcal{A}$.

If an interpretation $\mathcal{I}_{\mathcal{D}}$ is a model of a TBox $\mathcal{T}$, an RBox $\mathcal{R}$ or an ABox $\mathcal{A}$, then $\mathcal{I}_{\mathcal{D}}$ is also said to satisfy $\mathcal{T}$, $\mathcal{R}$ or $\mathcal{A}$, respectively.

Definition 9 (Knowledge Base) A knowledge base is a triple $(\mathcal{T}, \mathcal{R}, \mathcal{A})$ where $\mathcal{T}$ is a TBox, $\mathcal{R}$ is an RBox and $\mathcal{A}$ is an ABox. An interpretation $\mathcal{I}_\mathcal{D}$ is a *model of a knowledge base* $(\mathcal{T}, \mathcal{R}, \mathcal{A})$ iff $\mathcal{I}_\mathcal{D}$ is a model of $\mathcal{T}$, $\mathcal{R}$ and $\mathcal{A}$.

Definition 10 (Inference Problems, Consistency) A *concept* is called *consistent* (w.r.t. a TBox $\mathcal{T}$ and an RBox $\mathcal{R}$) iff there exists a model of C (that is also a model of $\mathcal{T}$ and $\mathcal{R}$). An *ABox* $\mathcal{A}$ is *consistent* (w.r.t. a TBox $\mathcal{T}$ and an RBox $\mathcal{R}$) iff $\mathcal{A}$ has model $\mathcal{I}_\mathcal{D}$ (which is also a model of $\mathcal{T}$ and $\mathcal{R}$). A *knowledge base* is called *consistent* iff there exists a model. A concept, ABox or knowledge base which is not consistent is called inconsistent.

3 Solving an Application Problem with $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$

According to [Buchheit et al., 1994; Buchheit et al., 1995] configuration problem solving processes can be formalized as synthesis inference tasks. Following this approach, a solution of a configuration task is defined to be a *logical model* of the given knowledge base consisting of both the conceptual domain model (TBox, RBox) as well as the task specification (ABox).[1] The TBox and the RBox describe the configuration space.

Note that specific languages for describing the configuration space might be used. For instance, BHIBS [Cunis, 1991] is a configuration frame language which allows one to describe the properties of instances by specifying restrictions for the required values of named slots. The values can be either single objects or sets of objects, and the restrictions can be specified extensionally by directly giving concrete values like numbers, symbols or instances of concepts, or by intensionally describing sets and sequences of objects. The following example of an expression of the BHIBS-language describes the concept of a cylinder:

```
(is! (a Cylinder)
     (a Motorpart
        (part-of (a Motor))
        (displacement [1ccm 1000ccm])
        (has-parts
           (:set #[(a Cylinderpart) 4 6] :=
                 #[(a Piston)       1 1]
                 #[(a Piston-Rod)   1 1]
                 #[(a Valve)        2 4]))))
```

A $\mathsf{Cylinder}$ is required to be a $\mathsf{Motorpart}$, to be $\mathsf{part\_of}$ a Motor, to have a $\mathsf{displacement}$ of 1 to 1000ccm, and to have a set of 4 to 6 parts ($\mathsf{has\_parts}$) which are all $\mathsf{Cylinderparts}$ and it consists of exactly 1 Piston, exactly 1

[1] [Buchheit et al., 1995] additionally considers relations defined with definite clauses.

Piston_Rod, and 2 to 4 Valves. This expression can be transformed to a terminological inclusion axiom of a description logic providing concrete domains. Let the concrete domain $\Re$ be defined as in [Baader and Hanschke, 1991b]: $\Re = (\mathbb{R}, \Phi_\Re)$ where $\Phi_\Re$ is a set of predicates which are based on polynomial equations or inequations. The concrete domain $\Re$ is admissible (see also [Baader and Hanschke, 1991b]).

The cylinder example is translated as follows (the term $\lambda_{\text{Vol}}\, c. (\dots)$ is a unary predicate of a numeric concrete domain for the dimension *Volume* with base unit m^3):

Using the language $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ this can be translated into description logics. First of all, a role box $\mathcal{R}$ is defined to contain the following role inclusion axioms.

$$
\begin{aligned}
\textbf{has\_cylinder\_part} &\sqsubseteq \mathsf{has\_part}\\
\textbf{has\_piston\_part} &\sqsubseteq \mathsf{has\_part}\\
\textbf{has\_piston\_rod\_part} &\sqsubseteq \mathsf{has\_part}\\
\textbf{has\_valve\_part} &\sqsubseteq \mathsf{has\_part}
\end{aligned}
$$

Then, a TBox $\mathcal{T}$ is given as a set of terminological axioms. For instance, the following range restrictions are declared.

$$
\begin{aligned}
\top &\sqsubseteq \forall\, \mathsf{has\_cylinder\_part} \,.\, \mathsf{Cylinder}\\
\top &\sqsubseteq \forall\, \mathsf{has\_piston\_part} \,.\, \mathsf{Piston}\\
\top &\sqsubseteq \forall\, \mathsf{has\_piston\_rod\_part} \,.\, \mathsf{Piston\_Rod}\\
\top &\sqsubseteq \forall\, \mathsf{has\_valve\_part} \,.\, \mathsf{Valve}
\end{aligned}
$$

For Cylinderpart a so-called cover axiom is given. Moreover, additional axioms ensure the disjointness of more specific subconcepts of Cylinderpart.

$$
\begin{aligned}
\textbf{Cylinderpart} &\sqsubseteq \mathsf{Piston} \sqcup \mathsf{Piston\_Rod} \sqcup \mathsf{Valve}\\
\textbf{Piston} &\sqsubseteq \neg\mathsf{Piston\_Rod} \sqcap \neg\mathsf{Valve}\\
\textbf{Piston\_Rod} &\sqsubseteq \neg\mathsf{Piston} \sqcap \neg\mathsf{Valve}\\
\textbf{Valve} &\sqsubseteq \neg\mathsf{Piston} \sqcap \neg\mathsf{Piston\_Rod}
\end{aligned}
$$

Now another axiom relates a Cylinder to its parts. We assume that displacement is declared as a feature.

$$\begin{aligned}\textsf{\textbf{Cylinder}} \sqsubseteq\ & \textsf{Motorpart} \sqcap \\ & \exists_{=1}\ \textsf{part\_of} \sqcap \\ & \exists\, \textsf{displacement} \,.\, \lambda_{\mathrm{Vol}}\, c \,.\, 0.001 \leq c \leq 1 \sqcap \\ & \forall\, \textsf{has\_parts} \,.\, \textsf{Cylinderpart} \sqcap \\ & \exists_{\geq 4}\ \textsf{has\_cylinder\_part} \sqcap \\ & \exists_{\leq 6}\ \textsf{has\_cylinder\_part} \sqcap \\ & \exists_{=1}\ \textsf{has\_piston\_part} \sqcap \\ & \exists_{=1}\ \textsf{has\_piston\_rod\_part} \sqcap \\ & \exists_{\geq 2}\ \textsf{has\_valve\_part} \sqcap \\ & \exists_{\leq 4}\ \textsf{has\_valve\_part}\end{aligned}$$

In our example, the ABox being used is very simple:

$\mathcal{A} = \{\textsf{a} : \textsf{Cylinder} \sqcap \exists\, \textsf{displacement} \,.\, \lambda_{\mathrm{Vol}}\, c \,.\, c \geq 0.5\}$.

In order to solve the configuration problem, the knowledge base $(\mathcal{T}, \mathcal{R}, \mathcal{A})$ is tested for consistency. If the knowledge base is consistent, there exists a model. The model can be interpreted as a solution to the configuration problem [Buchheit et al., 1994]. Note that $(\mathcal{T}, \mathcal{R}, \mathcal{A})$ is only a very simplified example for a representation of a configuration problem. For instance, using an ABox with additional assertions it is possible to explicitly specify some required cylinder parts etc.

In order to actually compute a solution to a configuration problem, a sound and complete calculus for the $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ knowledge base consistency problem is required that terminates on any input. If the calculus returns "consistent" then (parts of) the internal structures used in the proof can be printed as a problem solution in a convenient form. We will return to this point after the discussion of the tableaux calculus for $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$.

4 A Tableaux Calculus for $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$

In the following a calculus to decide the consistency of an $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ knowledge base $(\mathcal{T}, \mathcal{R}, \mathcal{A})$ is devised. As a first step the original ABox $\mathcal{A}$ of the knowledge base is transformed w.r.t. the TBox $\mathcal{T}$. The idea is to derive an ABox $\mathcal{A}_\mathcal{T}$ that is consistent w.r.t. an RBox $\mathcal{R}$ (and an empty TBox) iff $(\mathcal{T}, \mathcal{R}, \mathcal{A})$ is consistent. The calculus introduced below is applied to $\mathcal{A}_\mathcal{T}$ and the role box $\mathcal{R}$.

In order to define the transformation steps for deriving $\mathcal{A}_\mathcal{T}$, we have to introduce a few technical terms. First, for any concept term we define its negation normal form.

Definition 11 (Negation Normal Form) A concept is in negation normal form iff negation signs may occur only in front of concept names.

Proposition 12 Every $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ concept term C can be transformed into negation normal form $\mathit{nnf}(\mathsf{C})$ by recursively applying the following transformation rules to subconcepts from left to right. If no rule is applicable, the resulting concept is in negation normal form and all models of C are also models of $\mathit{nnf}(\mathsf{C})$ and vice versa. The transformation is possible in linear time.

- $\neg(\mathsf{C} \sqcap \mathsf{D}) \rightarrow \neg\mathsf{C} \sqcup \neg\mathsf{D}$
- $\neg(\mathsf{C} \sqcup \mathsf{D}) \rightarrow \neg\mathsf{C} \sqcap \neg\mathsf{D}$
- $\neg\forall\,\mathsf{R}\,.\,\mathsf{C} \rightarrow \exists\,\mathsf{R}\,.\,\neg\mathsf{C}$
- $\neg\exists\,\mathsf{R}\,.\,\mathsf{C} \rightarrow \forall\,\mathsf{R}\,.\,\neg\mathsf{C}$
- $\neg\exists_{\leq m}\,\mathsf{S} \rightarrow \exists_{\geq m+1}\,\mathsf{S}$
- $\neg\exists_{\geq m}\,\mathsf{S} \rightarrow \exists_{\leq m-1}\,\mathsf{S}$
- $\neg\exists\,\mathsf{f_1}, \ldots, \mathsf{f_n}\,.\,\mathsf{P} \rightarrow \exists\,\mathsf{f_1}, \ldots, \mathsf{f_n}\,.\,\overline{\mathsf{P}} \sqcup \forall\,\mathsf{f_1}\,.\,\bot_{\mathcal{D}} \sqcup \ldots \sqcup \forall\,\mathsf{f_n}\,.\,\bot_{\mathcal{D}}$
  where $\overline{\mathsf{P}}$ is the negation of P.
- $\neg\forall\,\mathsf{f}\,.\,\bot_{\mathcal{D}} \rightarrow \exists\,\mathsf{f}\,.\,\top_{\mathcal{D}}$

Definition 13 (Additional ABox Assertions) Let C be a concept term, $\mathsf{a}, \mathsf{b} \in O$ be individual names, and $x \notin O \cup O_C$, then the following expressions are also assertional axioms:

- $\forall\, x\,.\,x\!:\!\mathsf{C}$ *(universal concept assertion)*,[2]
- $\mathsf{a} \dot{\neq} \mathsf{b}$ *(inequality assertion)*.

An interpretation $\mathcal{I}_{\mathcal{D}}$ satisfies an assertional axiom $\forall\, x\,.\,x\!:\!\mathsf{C}$ iff $\mathsf{C}^{\mathcal{I}} = \Delta_{\mathcal{I}}$ and $\mathsf{a} \dot{\neq} \mathsf{b}$ iff $\mathsf{a}^{\mathcal{I}} \neq \mathsf{b}^{\mathcal{I}}$.

Definition 14 (Fork, Fork Elimination) If it holds that $\{(\mathsf{a}, \mathsf{x_1})\!:\!\mathsf{f}, (\mathsf{a}, \mathsf{x_2})\!:\!\mathsf{f}\} \subseteq \mathcal{A}$ then there exists a *fork* in $\mathcal{A}$. In case of a fork w.r.t. $\mathsf{x_1}, \mathsf{x_2}$, the replacement of every occurrence of $\mathsf{x_2}$ in $\mathcal{A}$ by $\mathsf{x_1}$ is called *fork elimination*.

We are now ready to define an augmented ABox as input to the tableaux rules.

Definition 15 (Augmented ABox) For an initial ABox $\mathcal{A}$ we define its *augmented* ABox $\mathcal{A}_{\mathcal{T}}$ w.r.t a TBox $\mathcal{T}$ by applying the following transformation rules to $\mathcal{A}$. First of all, all forks in $\mathcal{A}$ are eliminated. Then, for every GCI $\mathsf{C} \sqsubseteq \mathsf{D}$ in $\mathcal{T}$ the assertion $\forall\, x\,.\,x\!:\!(\neg\mathsf{C} \sqcup \mathsf{D})$ is added to $\mathcal{A}$. Every concept term occurring in $\mathcal{A}$ is transformed into its negation normal form. Let

[2] $\forall\, x\,.\,x\!:\!\mathsf{C}$ is to be read as $\forall\, x\,.\,(x\!:\!\mathsf{C})$.

$O_{\mathcal{A}} = \{a_1, \ldots, a_n\}$ be the set of individuals mentioned in $\mathcal{A}$, then the set of inequality assertions $\{a_i \neq a_j \mid a_i, a_j \in O_{\mathcal{A}},\ i, j \in 1..n, i \neq j\}$ is added to $\mathcal{A}$.

In order to check the consistency of an $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ knowledge base $(\mathcal{T}, \mathcal{R}, \mathcal{A})$ the augmented ABox $\mathcal{A}_{\mathcal{T}}$ is computed. Then, the tableaux rules are applied to the augmented ABox $\mathcal{A}_{\mathcal{T}}$ and a role box $\mathcal{R}$. The rules are applied in accordance with a completion strategy (see below).

Lemma 16 A knowledge base $(\mathcal{T}, \mathcal{R}, \mathcal{A})$ is consistent if and only if $\mathcal{A}_{\mathcal{T}}$ is consistent w.r.t. the role box $\mathcal{R}$ (and an empty TBox).

Proof. "$\Rightarrow$" Since $(\mathcal{T}, \mathcal{R}, \mathcal{A})$ is consistent there exists a model $\mathcal{I}_{\mathcal{D}} = (\Delta_{\mathcal{I}}, \Delta_{\mathcal{D}}, \cdot^{\mathcal{I}})$ such that $\forall\, C \sqsubseteq D \in \mathcal{T} : C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. This is equivalent to $\forall\, a \in \Delta_{\mathcal{I}} : a \in C^{\mathcal{I}} \implies a \in D^{\mathcal{I}}$. Hence, $\forall\, a \in \Delta_{\mathcal{I}} : a \in (\neg C)^{\mathcal{I}} \vee a \in D^{\mathcal{I}}$ or $\forall\, a \in \Delta_{\mathcal{I}} : a \in (\neg C \sqcup D)^{\mathcal{I}}$. In other words: $(\neg C \sqcup D)^{\mathcal{I}} = \Delta_{\mathcal{I}}$. Thus, due to the semantics defined above $\forall\, x \,.\, x : \neg C \sqcup D$ is also satisfied.

"$\Leftarrow$" This can be shown by applying the arguments in the other direction.

Since all forks are eliminated in $\mathcal{A}_{\mathcal{T}}$ and all terminological axioms in $\mathcal{T}$ are appropriately represented in $\mathcal{A}_{\mathcal{T}}$, a model for both $\mathcal{A}_{\mathcal{T}}$ and $\mathcal{R}$ is also a model for $\mathcal{A}$, $\mathcal{T}$ and $\mathcal{R}$ and vice versa. □

The tableaux rules require the notion of blocking their applicability. This is based on so-called concept sets, an ordering for new individuals and concrete objects, and the notion of a blocking individual.

Definition 17 (Concept Set, $\mathcal{A}$-equivalent) Given an ABox $\mathcal{A}$ and an individual a occurring in $\mathcal{A}$, we define the *concept set* of a as $\sigma(\mathcal{A}, a) := \{C \mid a : C \in \mathcal{A}\}$. We define two individuals as $\mathcal{A}$-*equivalent*, written $a \equiv_{\mathcal{A}} b$, if their concept sets are equal, i.e. $\sigma(\mathcal{A}, a) = \sigma(\mathcal{A}, b)$.

Definition 18 (Ordering) We define an *individual ordering* '$\prec$' for new individuals (elements of O_N) occurring in an ABox $\mathcal{A}$. If $b \in O_N$ is introduced in $\mathcal{A}$, then $a \prec b$ for all new individuals a already present in $\mathcal{A}$. A *concrete object ordering* '$\prec_C$' for elements of O_C occurring in an ABox $\mathcal{A}$ is defined as follows. If $y \in O_C$ is introduced in $\mathcal{A}$, then $x \prec_C y$ for all concrete objects x already present in $\mathcal{A}$.

Definition 19 (Blocking Individual, blocked) Let $\mathcal{A}$ be an ABox and $a, b \in O_N$ be individuals in $\mathcal{A}$. We call a the *blocking individual* of b if the following conditions hold:

1. $\sigma(\mathcal{A}, a) \supseteq \sigma(\mathcal{A}, b)$
2. $a \prec b$
3. $\neg \exists\, c$ in $\mathcal{A} : c \in O_N, c \prec a, \sigma(\mathcal{A}, c) \supseteq \sigma(\mathcal{A}, b)$.

If a is a blocking individual for b, then b is said to be *blocked* by a.

4.1 Completion Rules

We are now ready to define the *completion rules* that are intended to generate a so-called completion (see also below) of an ABox $\mathcal{A}_\mathcal{T}$ w.r.t. an RBox $\mathcal{R}$. From this point on, if we refer to an ABox $\mathcal{A}$, we always consider ABoxes derived from $\mathcal{A}_\mathcal{T}$.

Definition 20 (Completion Rules)

R$\sqcap$ The conjunction rule.
if 1. $\mathsf{a}:\mathsf{C}\sqcap\mathsf{D}\in\mathcal{A}$, and
2. $\{\mathsf{a}:\mathsf{C},\ \mathsf{a}:\mathsf{D}\}\not\subseteq\mathcal{A}$
then $\mathcal{A}'=\mathcal{A}\cup\{\mathsf{a}:\mathsf{C},\ \mathsf{a}:\mathsf{D}\}$

R$\sqcup$ The disjunction rule (nondeterministic).
if 1. $\mathsf{a}:\mathsf{C}\sqcup\mathsf{D}\in\mathcal{A}$, and
2. $\{\mathsf{a}:\mathsf{C},\ \mathsf{a}:\mathsf{D}\}\cap\mathcal{A}=\emptyset$
then $\mathcal{A}'=\mathcal{A}\cup\{\mathsf{a}:\mathsf{C}\}$ **or** $\mathcal{A}'=\mathcal{A}\cup\{\mathsf{a}:\mathsf{D}\}$

R$\forall$C The role value restriction rule.
if 1. $\mathsf{a}:\forall\,\mathsf{R}\,.\,\mathsf{C}\in\mathcal{A}$, and
2. $\exists\,\mathsf{b}\in O,\mathsf{S}\in\mathsf{R}^{\downarrow}:(\mathsf{a},\mathsf{b}):\mathsf{S}\in\mathcal{A}$, and
3. $\mathsf{b}:\mathsf{C}\notin\mathcal{A}$
then $\mathcal{A}'=\mathcal{A}\cup\{\mathsf{b}:\mathsf{C}\}$

R$\forall_+$C The transitive role value restriction rule.
if 1. $\mathsf{a}:\forall\,\mathsf{R}\,.\,\mathsf{C}\in\mathcal{A}$, and
2. $\exists\,\mathsf{b}\in O,\mathsf{T}\in\mathsf{R}^{\downarrow},\mathsf{T}\in T,\mathsf{S}\in\mathsf{T}^{\downarrow}:(\mathsf{a},\mathsf{b}):\mathsf{S}\in\mathcal{A}$, and
3. $\mathsf{b}:\forall\,\mathsf{T}\,.\,\mathsf{C}\notin\mathcal{A}$
then $\mathcal{A}'=\mathcal{A}\cup\{\mathsf{b}:\forall\,\mathsf{T}\,.\,\mathsf{C}\}$

R$\forall_x$ The universal concept restriction rule.
if 1. $\forall\,x\,.\,x:\mathsf{C}\in\mathcal{A}$, and
2. $\exists\,\mathsf{a}\in O$: a mentioned in $\mathcal{A}$, and
3. $\mathsf{a}:\mathsf{C}\notin\mathcal{A}$
then $\mathcal{A}'=\mathcal{A}\cup\{\mathsf{a}:\mathsf{C}\}$

R$\exists$C The role exists restriction rule (generating).
if 1. $\mathsf{a}:\exists\,\mathsf{R}\,.\,\mathsf{C}\in\mathcal{A}$, and
2. $\mathsf{a}\in O_N\Rightarrow(\neg\exists\,\mathsf{c}$ in $\mathcal{A}$: $\mathsf{c}\in O_N,\mathsf{c}$ is a blocking individual for $\mathsf{a})$, and
3. $\neg\exists\,\mathsf{b}\in O,\mathsf{S}\in\mathsf{R}^{\downarrow}:\{(\mathsf{a},\mathsf{b}):\mathsf{S},\ \mathsf{b}:\mathsf{C}\}\subseteq\mathcal{A}$
then $\mathcal{A}'=\mathcal{A}\cup\{(\mathsf{a},\mathsf{b}):\mathsf{R},\ \mathsf{b}:\mathsf{C}\}$ where $\mathsf{b}\in O_N$ is not used in $\mathcal{A}$

R$\exists_{\geq n}$ The number restriction exists rule (generating).

if 1. $\mathsf{a} : \exists_{\geq n} \mathsf{R} \in \mathcal{A}$, and

2. $\mathsf{a} \in O_N \Rightarrow (\neg\exists\, \mathsf{c}$ in $\mathcal{A} : \mathsf{c} \in O_N, \mathsf{c}$ is a blocking individual for $\mathsf{a})$, and

3. $\neg\exists\, \mathsf{b_1}, \ldots, \mathsf{b_n} \in O, \mathsf{S_1}, \ldots, \mathsf{S_n} \in \mathsf{R}^{\downarrow}$:
$\{(\mathsf{a}, \mathsf{b_k}) : \mathsf{S_k} \mid \mathsf{k} \in 1..\mathsf{n}\} \cup \{\mathsf{b_i} \neq \mathsf{b_j} \mid \mathsf{i}, \mathsf{j} \in 1..\mathsf{n}, \mathsf{i} \neq \mathsf{j}\} \subseteq \mathcal{A}$

then $\mathcal{A}' = \mathcal{A} \cup \{(\mathsf{a}, \mathsf{b_k}) : \mathsf{R} \mid \mathsf{k} \in 1..\mathsf{n}\} \cup \{\mathsf{b_i} \neq \mathsf{b_j} \mid \mathsf{i}, \mathsf{j} \in 1..\mathsf{n}, \mathsf{i} \neq \mathsf{j}\}$
where $\mathsf{b_1}, \ldots, \mathsf{b_n} \in O_N$ are not used in $\mathcal{A}$

R$\exists_{\leq n}$ The number restriction merge rule (nondeterministic).

if 1. $\mathsf{a} : \exists_{\leq n} \mathsf{R} \in \mathcal{A}$, and

2. $\exists\, \mathsf{b_1}, \ldots, \mathsf{b_m} \in O, \mathsf{S_1}, \ldots, \mathsf{S_m} \in \mathsf{R}^{\downarrow}$: $\{(\mathsf{a}, \mathsf{b_1}) : \mathsf{S_1}, \ldots, (\mathsf{a}, \mathsf{b_m}) : \mathsf{S_m}\} \subseteq \mathcal{A}$
with $\mathsf{m} > \mathsf{n}$, and

3. $\exists\, \mathsf{b_i}, \mathsf{b_j} \in \{\mathsf{b_1}, \ldots, \mathsf{b_m}\} : \mathsf{i} \neq \mathsf{j}$, $\mathsf{b_i} \neq \mathsf{b_j} \notin \mathcal{A}$

then $\mathcal{A}' = \mathcal{A}[\mathsf{b_i}/\mathsf{b_j}]$, i.e. replace every occurrence of $\mathsf{b_i}$ in $\mathcal{A}$ by $\mathsf{b_j}$

R$\exists$P The predicate exists rule (generating).

if 1. $\mathsf{a} : \exists\, \mathsf{f_1}, \ldots, \mathsf{f_n} \,.\, \mathsf{P} \in \mathcal{A}$, and

2. $\neg\exists \mathsf{x_1}, \ldots, \mathsf{x_n} \in \mathsf{O_C} : \{(\mathsf{a}, \mathsf{x_1}) : \mathsf{f_1}, \ldots (\mathsf{a}, \mathsf{x_n}) : \mathsf{f_n}, (\mathsf{x_1}, \ldots, \mathsf{x_n}) : \mathsf{P}\} \subseteq \mathcal{A}$

then $\mathcal{A}' = \mathcal{A} \cup \{(\mathsf{a}, \mathsf{x_1}) : \mathsf{f_1}, \ldots (\mathsf{a}, \mathsf{x_n}) : \mathsf{f_n}, (\mathsf{x_1}, \ldots, \mathsf{x_n}) : \mathsf{P}\}$
where $\mathsf{x_1}, \ldots, \mathsf{x_n} \in O_C$ are not used in $\mathcal{A}$,
eliminate all forks $\{(\mathsf{a}, \mathsf{x}) : \mathsf{f_i}, (\mathsf{a}, \mathsf{x_i}) : \mathsf{f_i}\} \subseteq \mathcal{A}$
such that $(\mathsf{a}, \mathsf{x}) : \mathsf{f_i}$ remains in $\mathcal{A}$ if $\mathsf{x} \prec_C \mathsf{x_i}, \mathsf{i} \in 1..\mathsf{n}$

We call the rules R$\sqcup$ and R$\exists_{\leq n}$ *nondeterministic* rules since they can be applied in different ways to the same ABox. The remaining rules are called *deterministic* rules. Moreover, we call the rules R$\exists$C, R$\exists_{\geq n}$ and R$\exists$P *generating* rules since they are rules that can introduce new individuals.

Proposition 21 (Invariance) Let $\mathcal{A}$ and $\mathcal{A}'$ be ABoxes and $\mathcal{R}$ be a role box. Then:

1. If $\mathcal{A}'$ is derived from $\mathcal{A}$ w.r.t. $\mathcal{R}$ by applying a deterministic rule, then $\mathcal{A}$ is consistent w.r.t. $\mathcal{R}$ iff $\mathcal{A}'$ is consistent w.r.t. $\mathcal{R}$.

2. If $\mathcal{A}'$ is derived from $\mathcal{A}$ w.r.t. $\mathcal{R}$ by applying a nondeterministic rule, then $\mathcal{A}$ is consistent w.r.t. $\mathcal{R}$ if $\mathcal{A}'$ is consistent w.r.t. $\mathcal{R}$. Conversely, if $\mathcal{A}$ is consistent w.r.t. $\mathcal{R}$ and a nondeterministic rule is applicable to $\mathcal{A}$, then it can be applied in such a way that it yields an ABox $\mathcal{A}'$ consistent w.r.t. $\mathcal{R}$.

Proof.

1. "$\Leftarrow$" Due to the structure of the deterministic rules one can immediately verify that $\mathcal{A}$ is a subset of $\mathcal{A}'$. Therefore, $\mathcal{A}$ is consistent w.r.t. $\mathcal{R}$ if $\mathcal{A}'$ is consistent w.r.t. $\mathcal{R}$.

"$\Rightarrow$" In order to show that $\mathcal{A}'$ is consistent w.r.t. $\mathcal{R}$ after applying a deterministic rule to the consistent ABox $\mathcal{A}$, we examine each applicable rule separately. We assume that $\mathcal{I}_\mathcal{D} = (\Delta_\mathcal{I}, \Delta_\mathcal{D}, \cdot^\mathcal{I})$ satisfies $\mathcal{A}$ and $\mathcal{R}$. Then, we observe that it is an obvious consequence that $\mathsf{R}^\mathcal{I} \subseteq \mathsf{S}^\mathcal{I}$ iff $(\mathsf{R}, \mathsf{S}) \in \sqsubseteq^*_\mathcal{R}$.

If the conjunction rule is applied to $\mathsf{a} : \mathsf{C} \sqcap \mathsf{D} \in \mathcal{A}$, then we get a new Abox $\mathcal{A}' = \mathcal{A} \cup \{\mathsf{a} : \mathsf{C},\ \mathsf{a} : \mathsf{D}\}$. Since $\mathcal{I}_\mathcal{D}$ satisfies $\mathsf{a} : \mathsf{C} \sqcap \mathsf{D}$, $\mathcal{I}_\mathcal{D}$ satisfies $\mathsf{a} : \mathsf{C}$ and $\mathsf{a} : \mathsf{D}$ and therefore $\mathcal{A}'$.

If the role value restriction rule is applied to $\mathsf{a} : \forall\, \mathsf{R} \,.\, \mathsf{C} \in \mathcal{A}$, then there must be a role assertion $(\mathsf{a}, \mathsf{b}) : \mathsf{S} \in \mathcal{A}$ with $\mathsf{S} \in \mathsf{R}^\downarrow$ such that $\mathcal{A}' = \mathcal{A} \cup \{\mathsf{b} : \mathsf{C}\}$. Since $\mathcal{I}_\mathcal{D}$ satisfies $\mathcal{A}$ and $\mathcal{R}$, it holds that $(\mathsf{a}^\mathcal{I}, \mathsf{b}^\mathcal{I}) \in \mathsf{S}^\mathcal{I}, \mathsf{S}^\mathcal{I} \subseteq \mathsf{R}^\mathcal{I}$. Since $\mathcal{I}_\mathcal{D}$ satisfies $\mathsf{a} : \forall\, \mathsf{R} \,.\, \mathsf{C}$, it holds that $\mathsf{b}^\mathcal{I} \in \mathsf{C}^\mathcal{I}$. Thus, $\mathcal{I}_\mathcal{D}$ satisfies $\mathsf{b} : \mathsf{C}$ and therefore $\mathcal{A}'$.

If the transitive role value restriction rule is applied to $\mathsf{a} : \forall\, \mathsf{R} \,.\, \mathsf{C} \in \mathcal{A}$, there must be an assertion $(\mathsf{a}, \mathsf{b}) : \mathsf{S} \in \mathcal{A}$ with $\mathsf{S} \in \mathsf{T}^\downarrow$ for some $\mathsf{T} \in T$ and $\mathsf{T} \in \mathsf{R}^\downarrow$ such that we get $\mathcal{A}' = \mathcal{A} \cup \{\mathsf{b} : \forall\, \mathsf{T} \,.\, \mathsf{C}\}$. Since $\mathcal{I}_\mathcal{D}$ satisfies $\mathcal{A}$ and $\mathcal{R}$, we have $\mathsf{a}^\mathcal{I} \in (\forall\, \mathsf{R} \,.\, \mathsf{C})^\mathcal{I}$ and $(\mathsf{a}^\mathcal{I}, \mathsf{b}^\mathcal{I}) \in \mathsf{S}^\mathcal{I}, \mathsf{S}^\mathcal{I} \subseteq \mathsf{T}^\mathcal{I} \subseteq \mathsf{R}^\mathcal{I}$. Since $\mathsf{a}^\mathcal{I} \in (\forall\, \mathsf{T} \,.\, \mathsf{C})^\mathcal{I}$, $\mathcal{I}_\mathcal{D}$ satisfies $\mathsf{a} : \forall\, \mathsf{T} \,.\, \mathsf{C}$ and $\mathsf{T} \in T, \mathsf{T} \in \mathsf{R}^\downarrow$, it holds that $\mathsf{b}^\mathcal{I} \in (\forall\, \mathsf{T} \,.\, \mathsf{C})^\mathcal{I}$ unless there exists a successor c of b such that $(\mathsf{b}, \mathsf{c}) : \mathsf{S}' \in \mathcal{A}$, $(\mathsf{b}^\mathcal{I}, \mathsf{c}^\mathcal{I}) \in \mathsf{S}'^\mathcal{I} \subseteq \mathsf{T}^\mathcal{I}$ and $\mathsf{c}^\mathcal{I} \notin \mathsf{C}^\mathcal{I}$. It follows from $(\mathsf{a}^\mathcal{I}, \mathsf{b}^\mathcal{I}) \in \mathsf{T}^\mathcal{I}$, $(\mathsf{b}^\mathcal{I}, \mathsf{c}^\mathcal{I}) \in \mathsf{T}^\mathcal{I}$, and $\mathsf{T} \in T$ that $(\mathsf{a}^\mathcal{I}, \mathsf{c}^\mathcal{I}) \in \mathsf{T}^\mathcal{I}, \mathsf{T}^\mathcal{I} \subseteq \mathsf{R}^\mathcal{I}$ and $\mathsf{a}^\mathcal{I} \notin (\forall\, \mathsf{R} \,.\, \mathsf{C})^\mathcal{I}$ in contradiction to the assumption. Thus, $\mathcal{I}_\mathcal{D}$ satisfies $\mathsf{b} : \forall\, \mathsf{T} \,.\, \mathsf{C}$ and therefore $\mathcal{A}'$.

If the universal concept restriction rule is applied to an individual a in $\mathcal{A}$ because of $\forall\, x \,.\, x : \mathsf{C} \in \mathcal{A}$, then $\mathcal{A}' = \mathcal{A} \cup \{\mathsf{a} : \mathsf{C}\}$. Since $\mathcal{I}_\mathcal{D}$ satisfies $\mathcal{A}$ and $\mathcal{R}$, it holds that $\mathsf{C}^\mathcal{I} = \Delta_\mathcal{I}$. Thus, it holds that $\mathsf{a}^\mathcal{I} \in \mathsf{C}^\mathcal{I}$ and $\mathcal{I}_\mathcal{D}$ satisfies $\mathcal{A}'$.

If the role exists restriction rule is applied to $\mathsf{a} : \exists\, \mathsf{R} \,.\, \mathsf{C} \in \mathcal{A}$, then we get the ABox $\mathcal{A}' = \mathcal{A} \cup \{(\mathsf{a}, \mathsf{b}) : \mathsf{R},\ \mathsf{b} : \mathsf{C}\}$. Since $\mathcal{I}_\mathcal{D}$ satisfies $\mathcal{A}$ and $\mathcal{R}$, there exists a $y \in \Delta_\mathcal{I}$ such that $(\mathsf{a}^\mathcal{I}, y) \in \mathsf{R}^\mathcal{I}$ and $y \in \mathsf{C}^\mathcal{I}$. We define the interpretation function $\cdot^{\mathcal{I}'}$ such that $\mathsf{b}^{\mathcal{I}'} := y$ and $x^{\mathcal{I}'} := x^\mathcal{I}$ for $x \neq \mathsf{b}$. It is easy to show that $\mathcal{I}'_\mathcal{D} = (\Delta_\mathcal{I}, \Delta_\mathcal{D}, \cdot^{\mathcal{I}'})$ satisfies $\mathcal{A}'$.

If the number restriction exists rule is applied to $\mathsf{a} : \exists_{\geq n}\, \mathsf{R} \in \mathcal{A}$, then we get $\mathcal{A}' = \mathcal{A} \cup \{(\mathsf{a}, \mathsf{b_k}) : \mathsf{R} \mid \mathsf{k} \in 1..\mathsf{n}\} \cup \{\mathsf{b_i} \not\doteq \mathsf{b_j} \mid \mathsf{i}, \mathsf{j} \in 1..\mathsf{n}, \mathsf{i} \neq \mathsf{j}\}$. Since $\mathcal{I}_\mathcal{D}$ satisfies $\mathcal{A}$ and $\mathcal{R}$, there must exist n distinct individuals $y_i \in \Delta_\mathcal{I}$, $i \in 1..n$ such that $(\mathsf{a}^\mathcal{I}, y_i) \in \mathsf{R}^\mathcal{I}$. We define the interpretation function $\cdot^{\mathcal{I}'}$ such that $\mathsf{b_i}^{\mathcal{I}'} := y_i$ and $x^{\mathcal{I}'} := x^\mathcal{I}$ for $x \notin \{\mathsf{b_1}, \ldots, \mathsf{b_n}\}$. It is easy to show that $\mathcal{I}'_\mathcal{D} = (\Delta_\mathcal{I}, \Delta_\mathcal{D}, \cdot^{\mathcal{I}'})$ satisfies $\mathcal{A}'$.

If the predicate exists rule is applied to $\mathsf{a} : \exists\, \mathsf{f_1}, \ldots, \mathsf{f_n} \,.\, \mathsf{P} \in \mathcal{A}$, then we get the ABox $\mathcal{A}' = \mathcal{A} \cup \{(\mathsf{x_1}, \ldots, \mathsf{x_n}) : \mathsf{P}, (\mathsf{a}, \mathsf{x_1}) : \mathsf{f_1}, \ldots, (\mathsf{a}, \mathsf{x_n}) : \mathsf{f_n}\}$. After fork elimination, some $\mathsf{x_i}$ may be replaced by $\mathsf{z_i}$ with $\mathsf{z_i} \prec_C \mathsf{x_i}$. Since $\mathcal{I}_\mathcal{D}$ satisfies $\mathcal{A}$ and $\mathcal{R}$, there exist $y_1, \ldots, y_n \in \Delta_\mathcal{D}$ such that $\forall i \in \{1, \ldots, n\} : (\mathsf{a}^\mathcal{I}, y_i) \in \mathsf{f_i}^\mathcal{I}$ and $(y_1, \ldots, y_n) \in \mathsf{P}^\mathcal{I}$. We define the interpretation function $\cdot^{\mathcal{I}'}$ such that $\mathsf{x_i}^{\mathcal{I}'} := y_i$ for all $\mathsf{x_i}$ not replaced by $\mathsf{z_i}$ and $(y_1, \ldots, y_n) \in \mathsf{P}^{\mathcal{I}'}$. The fork elimination strategy used in the R∃P rule guarantees that concrete objects introduced in previous steps are not eliminated. Thus, it is ensured that the interpretation of $\mathsf{x_i}$ is not changed in $\mathcal{I}'_\mathcal{D}$. It is easy to see that $\mathcal{I}'_\mathcal{D} = (\Delta_\mathcal{I}, \Delta_\mathcal{D}, \cdot^{\mathcal{I}'})$ satisfies $\mathcal{A}'$.

2. "$\Leftarrow$" Assume that $\mathcal{A}'$ is satisfied by $\mathcal{I}'_\mathcal{D} = (\Delta_\mathcal{I}, \Delta_\mathcal{D}, \cdot^{\mathcal{I}'})$. By examining the nondeterministic rules we show that $\mathcal{A}$ is also consistent w.r.t. $\mathcal{R}$.

If $\mathcal{A}'$ is obtained from $\mathcal{A}$ by applying the disjunction rule, then $\mathcal{A}$ is a subset of $\mathcal{A}'$ and therefore satisfied by $\mathcal{I}'_\mathcal{D}$ and $\mathcal{R}$.

If $\mathcal{A}'$ is obtained from $\mathcal{A}$ by applying the number restriction merge rule to $\mathsf{a} : \exists_{\leq n} \mathsf{R} \in \mathcal{A}$, then there exist $\mathsf{b_i}, \mathsf{b_j}$ in $\mathcal{A}$ such that $\mathcal{A}' = \mathcal{A}[\mathsf{b_i}/\mathsf{b_j}]$. We define the interpretation function $\cdot^\mathcal{I}$ such that $\mathsf{b_i}^\mathcal{I} := \mathsf{b_j}^{\mathcal{I}'}$ and $\mathsf{x}^\mathcal{I} := \mathsf{x}^{\mathcal{I}'}$ for every $\mathsf{x} \neq \mathsf{b_i}$. Obviously, $\mathcal{I}_\mathcal{D} = (\Delta_\mathcal{I}, \Delta_\mathcal{D}, \cdot^\mathcal{I})$ satisfies $\mathcal{A}$ and $\mathcal{R}$.

"$\Rightarrow$" We suppose that $\mathcal{I}_\mathcal{D} = (\Delta_\mathcal{I}, \Delta_\mathcal{D}, \cdot^\mathcal{I})$ satisfies $\mathcal{A}$ and $\mathcal{R}$ and a nondeterministic rule is applicable to an individual a in $\mathcal{A}$.

If the disjunction rule is applicable to $\mathsf{a} : \mathsf{C} \sqcup \mathsf{D} \in \mathcal{A}$ and $\mathcal{A}$ is consistent w.r.t. $\mathcal{R}$, it holds $\mathsf{a}^\mathcal{I} \in (\mathsf{C} \sqcup \mathsf{D})^\mathcal{I}$. It follows that either $\mathsf{a}^\mathcal{I} \in \mathsf{C}^\mathcal{I}$ or $\mathsf{a}^\mathcal{I} \in \mathsf{D}^\mathcal{I}$ (or both). Hence, the disjunction rule can be applied in a way that $\mathcal{I}_\mathcal{D}$ also satisfies the ABox $\mathcal{A}'$.

If the number restriction merge rule is applicable to $\mathsf{a} : \exists_{\leq n} \mathsf{R} \in \mathcal{A}$ and $\mathcal{A}$ is consistent w.r.t. $\mathcal{R}$, it holds $\mathsf{a}^\mathcal{I} \in (\exists_{\leq n} \mathsf{R})^\mathcal{I}$ and $\|\{b \mid (a, b) \in \mathsf{R}^\mathcal{I}\}\| \leq n$. However, it also holds $\|\{\mathsf{b} \mid (\mathsf{a}^\mathcal{I}, \mathsf{b}^\mathcal{I}) \in \mathsf{R}^\mathcal{I}\}\| > m$ with $m \geq n$. Without loss of generality we only need to consider the case that $m = n + 1$. Thus, we can conclude by the Pigeonhole Principle that there exist at least two R-successors $\mathsf{b_i}, \mathsf{b_j}$ of a such that $\mathsf{b_i}^\mathcal{I} = \mathsf{b_j}^\mathcal{I}$. Since $\mathcal{I}_\mathcal{D}$ satisfies $\mathcal{A}$ and $\mathcal{R}$, at least one of the two individuals must be a new individual. Let us assume $\mathsf{b_i} \in O_N$, then $\mathcal{I}_\mathcal{D}$ obviously satisfies $\mathcal{A}[\mathsf{b_i}/\mathsf{b_j}]$ and $\mathcal{R}$. □

Given an ABox $\mathcal{A}$, more than one rule might be applicable to $\mathcal{A}$. This is controlled by a completion strategy in accordance to the ordering for new individuals (see Definition 18).

Definition 22 (Completion Strategy) We define a *completion strategy* that must observe the following restrictions.

- Meta rules:
  - Apply a rule to an individual $\mathsf{b} \in O_N$ only if no rule is applicable to an individual $\mathsf{a} \in O_O$.
  - Apply a rule to an individual $\mathsf{b} \in O_N$ only if no rule is applicable to another individual $\mathsf{a} \in O_N$ such that $\mathsf{a} \prec \mathsf{b}$.
- The completion rules are always applied in the following order. A step is skipped in case the corresponding set of applicable rules is empty.

1. Apply all nongenerating rules (R$\sqcap$, R$\sqcup$, R$\forall$C, R$\forall_+$C, R$\forall_x$, R$\exists_{\leq n}$) as long as possible.
2. Apply a generating rule (R$\exists$C, R$\exists_{\geq n}$, R$\exists$P) and restart with step 1 as long as possible.

In the following we always assume that rules are applied in accordance to this strategy. It ensures that the rules are applied to new individuals w.r.t. the ordering '$\prec$' which guarantees a breadth-first order. The application of rules stops immediately and backtracks to (possibly) remaining choice points, if a so-called clash is discovered.

Definition 23 (Clash,Clash Triggers, Completion) We assume the same naming conventions as used above. An ABox $\mathcal{A}$ contains a *clash* if one of the following *clash triggers* is applicable. If none of the clash triggers is applicable to $\mathcal{A}$, then $\mathcal{A}$ is called *clash-free*.

- *Primitive clash*: $\{\mathsf{a}:\mathsf{C}, \mathsf{a}:\neg\mathsf{C}\} \subseteq \mathcal{A}$
- *Number restriction merging clash*:
  $\exists \mathsf{S}_1, \ldots, \mathsf{S}_\mathsf{m} \in \mathsf{R}^{\downarrow} : \{\mathsf{a}:\exists_{\leq n}\,\mathsf{R}\} \cup \{(\mathsf{a},\mathsf{b}_\mathsf{i}):\mathsf{S}_\mathsf{i} \mid \mathsf{i} \in 1..\mathsf{m}\} \cup$
  $\{\mathsf{b}_\mathsf{i} \neq \mathsf{b}_\mathsf{j} \mid \mathsf{i},\mathsf{j} \in 1..\mathsf{m}, \mathsf{i} \neq \mathsf{j}\} \subseteq \mathcal{A}$ with $\mathsf{m} > \mathsf{n}$
- *No concrete domain feature clash*: $\{(\mathsf{a},\mathsf{x}):\mathsf{f}, \mathsf{a}:\forall\,\mathsf{f}\,.\,\bot_{\mathcal{D}}\} \subseteq \mathcal{A}$.
- *Concrete domain predicate clash*: $(\mathsf{x}_1^{(1)}, \ldots, \mathsf{x}_\mathsf{n}^{(1)}):\mathsf{P}_1 \in \mathcal{A}, \ldots,$
  $(\mathsf{x}_1^{(\mathsf{k})}, \ldots, \mathsf{x}_{\mathsf{n}_\mathsf{k}}^{(\mathsf{k})}):\mathsf{P}_\mathsf{k} \in \mathcal{A}$ and the conjunction $\bigwedge_{\mathsf{i}=1}^{\mathsf{k}} \mathsf{P}_\mathsf{i}(\mathsf{x}_1^{(\mathsf{i})}, \ldots, \mathsf{x}_{\mathsf{n}_\mathsf{i}}^{(\mathsf{i})})$ is not satisfiable in $\mathcal{D}$. Note that this can be decided since $\mathcal{D}$ is required to be admissible.

A clash-free ABox $\mathcal{A}$ is called *complete* if no completion rule is applicable to $\mathcal{A}$. A complete ABox $\mathcal{A}'$ derived from an ABox $\mathcal{A}$ is also called a *completion* of $\mathcal{A}$.

Any ABox containing a clash is obviously unsatisfiable (w.r.t. an RBox $\mathcal{R}$). The purpose of the calculus is to generate a completion for an initial ABox $\mathcal{A}_\mathcal{T}$ that proves the consistency of $\mathcal{A}_\mathcal{T}$ (w.r.t. an RBox $\mathcal{R}$) or its inconsistency if no completion can be found.

4.2 Decidability of the $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ ABox Consistency Problem

The following lemma proves that whenever a generating rule has been applied to an individual $\mathsf{a} \in O_N$, the concept set $\sigma(\cdot, \mathsf{a})$ of a does not change for succeeding ABoxes. Note that the original ABox does not contain elements from O_N (see Definition 8).

Lemma 24 (Stability) Let $\mathcal{A}$ be an ABox and $\mathsf{a} \in O_N$ be in $\mathcal{A}$. Let a generating rule be applicable to a according to the completion strategy. Let $\mathcal{A}'$ be any ABox derivable from $\mathcal{A}$ by any (possibly empty) sequence of rule applications. Then:

1. No rule is applicable in $\mathcal{A}'$ to an individual $\mathsf{b} \in O_N$ with $\mathsf{b} \prec \mathsf{a}$
2. $\sigma(\mathcal{A}, \mathsf{a}) = \sigma(\mathcal{A}', \mathsf{a})$, i.e. the concept set of a remains unchanged in $\mathcal{A}'$.
3. If $\mathsf{b} \in O_N$ is in $\mathcal{A}$ with $\mathsf{b} \prec \mathsf{a}$ then b is an individual in $\mathcal{A}'$, i.e. the individual b is not substituted by another individual.

Proof. Since in the original input ABox no elements of O_N are mentioned, a rule must have been applied if $\mathsf{b} \prec \mathsf{a}$ holds. **1.** By contradiction: Suppose $\mathcal{A} = \mathcal{A}_0 \rightarrow_* \cdots \rightarrow_* \mathcal{A}_n = \mathcal{A}'$, where $*$ is element of the completion rules and a rule is applicable to an individual b with $\mathsf{b} \prec \mathsf{a}$ in $\mathcal{A}'$. Then there has to exist a minimal i with $i \in 1..n$ such that this rule is also applicable in $\mathcal{A}_i$. If a rule is applicable to a in $\mathcal{A}$ then no rule is applicable to b in $\mathcal{A}$ due to our strategy. So no rule is applicable to any individual c such that $\mathsf{c} \prec \mathsf{a}$ in $\mathcal{A}_0, \ldots, \mathcal{A}_{i-1}$. It follows that from $\mathcal{A}_{i-1}$ to $\mathcal{A}_i$ a rule is applied to a or to a d such that $\mathsf{a} \prec \mathsf{d}$. Using an exhaustive case analysis of all rules we can show that no new assertion of the form $\mathsf{b} : \mathsf{C}$ or $(\mathsf{b}, \mathsf{e}) : \mathsf{R}$ can be added to $\mathcal{A}_{i-1}$. Therefore, no rule is applicable to b in $\mathcal{A}_i$. This is a contradiction to our assumption.

2. By contradiction: Suppose $\sigma(\mathcal{A}, \mathsf{a}) \neq \sigma(\mathcal{A}', \mathsf{a})$. Let b be the direct predecessor of a with $\mathsf{b} \prec \mathsf{a}$. A rule must have been applied to a and not to b because of point 1. Due to our strategy only generating rules are applicable to a that cannot add new elements to $\sigma(\cdot, \mathsf{a})$. This is an obvious contradiction.

3. This follows from point 1 and the completion strategy. $\square$

The next lemma guarantees the uniqueness of a blocking individual for a blocked individual. This is a precondition for defining a particular interpretation from $\mathcal{A}$.

Lemma 25 Let $\mathcal{A}'$ be an ABox and a be a new individual in $\mathcal{A}'$. If a is blocked then

1. a has no direct successor (individual from O) and
2. a has exactly one blocking individual.

Proof. **1.** By contradiction: Suppose that a is blocked in $\mathcal{A}'$ and $(\mathsf{a}, \mathsf{b}) : \mathsf{R} \in \mathcal{A}'$. There must exist an ancestor ABox $\mathcal{A}$ where a generating rule has been applied to a in $\mathcal{A}$. It follows from the definition of the generating rules that for every new individual c with $\mathsf{c} \prec \mathsf{a}$ in $\mathcal{A}$ we had $\sigma(\mathcal{A}, \mathsf{c}) \not\supseteq \sigma(\mathcal{A}, \mathsf{a})$. Since $\mathcal{A}'$

has been derived from $\mathcal{A}$ we can use Lemma 24 and conclude that for every new individual c with $\mathsf{c} \prec \mathsf{a}$ in $\mathcal{A}'$ we also have $\sigma(\mathcal{A}', \mathsf{c}) \not\supseteq \sigma(\mathcal{A}', \mathsf{a})$. Thus there cannot exist a blocking individual c for a in $\mathcal{A}'$. This is a contradiction to our hypothesis.

2. This follows directly from condition 3 in Definition 19. □

Definition 26 Let $\mathcal{A}$ be a complete ABox that has been derived by the calculus from an augmented ABox $\mathcal{A}_\mathcal{T}$ w.r.t. the role box $\mathcal{R}$. Since $\mathcal{A}$ is clash-free, there exists a variable assignment α that satisfies (the conjunction of) all occurring assertions $(\mathsf{x}_1, \ldots, \mathsf{x}_\mathsf{n}) : \mathsf{P} \in \mathcal{A}$. We define the *canonical interpretation* $\mathcal{I}_\mathcal{C} = (\Delta_{\mathcal{I}_\mathcal{C}}, \Delta_\mathcal{D}, \cdot^{\mathcal{I}_\mathcal{C}})$ w.r.t. $\mathcal{A}$ and $\mathcal{R}$ as follows:

1. $\Delta_{\mathcal{I}_\mathcal{C}} := \{\mathsf{a} \mid \mathsf{a} \text{ is an individual in } \mathcal{A}\}$
2. $\mathsf{a}^{\mathcal{I}_\mathcal{C}} := \mathsf{a}$ iff a is mentioned in $\mathcal{A}$
3. $\mathsf{x}^{\mathcal{I}_\mathcal{C}} := \alpha(\mathsf{x})$ iff x is mentioned in $\mathcal{A}$
4. $\mathsf{a} \in \mathsf{A}^{\mathcal{I}_\mathcal{C}}$ iff $\mathsf{a} : \mathsf{A} \in \mathcal{A}$
5. $(\mathsf{a}, \alpha(\mathsf{x})) \in \mathsf{f}^{\mathcal{I}_\mathcal{C}}$ iff $(\mathsf{a}, \mathsf{x}) : \mathsf{f} \in \mathcal{A}$
6. $(\mathsf{a}, \mathsf{b}) \in \mathsf{R}^{\mathcal{I}_\mathcal{C}}$ iff $\exists\, \mathsf{c}_0, \ldots, \mathsf{c}_\mathsf{n}$, $\mathsf{d}_0, \ldots, \mathsf{d}_{\mathsf{n}-1}$ mentioned in $\mathcal{A}$:[3],
   (a) $\mathsf{n} \geq 1, \mathsf{c}_0 = \mathsf{a}, \mathsf{c}_\mathsf{n} = \mathsf{b}$, and
   (b) $(\mathsf{a}, \mathsf{c}_1) : \mathsf{S}_1$, $(\mathsf{d}_1, \mathsf{c}_2) : \mathsf{S}_2$, $\ldots$ $(\mathsf{d}_{\mathsf{n}-2}, \mathsf{c}_{\mathsf{n}-1}) : \mathsf{S}_{\mathsf{n}-1}$, $(\mathsf{d}_{\mathsf{n}-1}, \mathsf{b}) : \mathsf{S}_\mathsf{n} \in \mathcal{A}$, and
   (c) $\forall \mathsf{i} \in 0..\mathsf{n}-1$:
   $\mathsf{d}_\mathsf{i} = \mathsf{c}_\mathsf{i}$ or
   d_i is a blocking individual for c_i, and $(\mathsf{d}_\mathsf{i}, \mathsf{c}_{\mathsf{i}+1}) : \mathsf{S}_{\mathsf{i}+1} \in \mathcal{A}$, and
   (d) if $\mathsf{n} > 1$
   $\forall\, \mathsf{i} \in 1..\mathsf{n} : \exists \mathsf{R}' \in T$, $\mathsf{R}' \in \mathsf{R}^{\downarrow}$, $\mathsf{S}_\mathsf{i} \in \mathsf{R}'^{\downarrow}$
   else
   $\mathsf{S}_1 \in \mathsf{R}^{\downarrow}$.

The construction of the canonical interpretation for the case 6 is illustrated with two examples in Figure 1. The following cases can be seen as special cases of case 6 introduced above ($\mathsf{n} = 1, \mathsf{c}_0 = \mathsf{a}, \mathsf{c}_1 = \mathsf{b}$):

- $\mathsf{c}_0 = \mathsf{d}_0$: $(\mathsf{a}, \mathsf{b}) \in \mathsf{R}^{\mathcal{I}_\mathcal{C}}$ iff $(\mathsf{c}_0, \mathsf{c}_1) : \mathsf{S}_1 \in \mathcal{A}$ for a role $\mathsf{S}_1 \in \mathsf{R}^{\downarrow}$.
- $\mathsf{c}_0 \neq \mathsf{d}_0$: $(\mathsf{a}, \mathsf{b}) \in \mathsf{R}^{\mathcal{I}_\mathcal{C}}$ iff d_0 is a blocking individual for c_0, and $(\mathsf{d}_0, \mathsf{c}_1) : \mathsf{S}_1 \in \mathcal{A}$, for a role $\mathsf{S}_1 \in \mathsf{R}^{\downarrow}$.

[3] Note that the variables $\mathsf{c}_0, \ldots, \mathsf{c}_\mathsf{n}, \mathsf{d}_0, \ldots, \mathsf{d}_{\mathsf{n}-1}$ not necessarily denote different individual names.

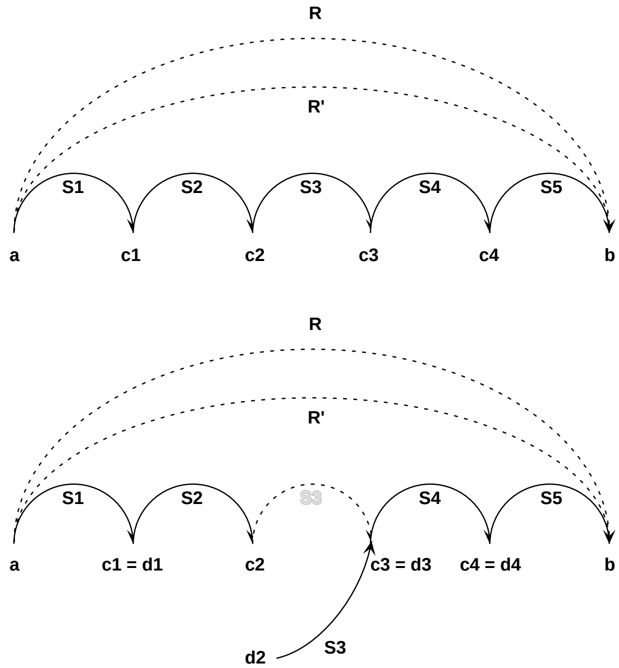

Figure 1: Construction of the canonical interpretation (two examples for case 6). In the lower example we assume that the individual $\mathsf{d2}$ is a blocking individual for $\mathsf{c2}$ (see text).

Due to Lemma 25, the canonical interpretation is well-defined because there exists a unique blocking individual for each individual that is blocked.

Theorem 27 (Soundness) Let $\mathcal{A}$ be a complete ABox that has been derived by the calculus from an augmented ABox $\mathcal{A}_{\mathcal{T}}$ w.r.t. the role box $\mathcal{R}$, then $\mathcal{A}_{\mathcal{T}}$ has a model which also satisfies all role axioms in $\mathcal{R}$.

Proof. Let $\mathcal{I}_{\mathcal{C}} = (\Delta_{\mathcal{I}_{\mathcal{C}}}, \Delta_{\mathcal{D}}, \cdot^{\mathcal{I}_{\mathcal{C}}})$ be the canonical interpretation for the ABox $\mathcal{A}$ constructed w.r.t. the TBox $\mathcal{T}$. $\mathcal{A}$ is clash-free.

Features are interpreted in the correct way: There can be no forks in $\mathcal{A}$ because (i) there are no forks in the augmented ABox $\mathcal{A}_{\mathcal{T}}$ and (ii) forks are immediately eliminated after an application of the R∃P rule. This rule is the only rule that introduces new assertions of the form $(\mathsf{a}, \mathsf{x}) : \mathsf{f} \in \mathcal{A}$. Note that forks cannot be introduced by the $\text{R}\exists_{\leq n}$ rule due to the completion strategy. Thus, $\mathcal{I}_{\mathcal{C}}$ maps features to (partial) functions because the variable assignment α is a function.

All role inclusions in the RBox $\mathcal{R}$ are satisfied: For every $\mathsf{S} \sqsubseteq \mathsf{R}$ in $\mathcal{R}$ it holds that $\mathsf{S}^{\mathcal{I}_{\mathcal{C}}} \subseteq \mathsf{R}^{\mathcal{I}_{\mathcal{C}}}$ This can be shown as follows. If $(\mathsf{a}^{\mathcal{I}_{\mathcal{C}}}, \mathsf{b}^{\mathcal{I}_{\mathcal{C}}}) \in S^{\mathcal{I}_{\mathcal{C}}}$, case 6 of Definition 26 must be applicable. Hence, there exists a chain of subroles possibly with gaps and blocking individuals (see Definition 26, case

6). Thus, the corresponding construction for $\mathcal{I}_\mathcal{C}$ adding $(\mathsf{a}^{\mathcal{I}_\mathcal{C}}, \mathsf{b}^{\mathcal{I}_\mathcal{C}})$ to $\mathsf{S}^{\mathcal{I}_\mathcal{C}}$ is also applicable to R since $\mathsf{S} \in \mathsf{R}^{\downarrow}$ (see 6d). Therefore, there is also tuple $(\mathsf{a}^{\mathcal{I}_\mathcal{C}}, \mathsf{b}^{\mathcal{I}_\mathcal{C}}) \in R^{\mathcal{I}_\mathcal{C}}$.

All transitivity axioms in the RBox $\mathcal{R}$ are satisfied, i.e. transitive roles are interpreted in the correct way: $\forall\, \mathsf{transitive}(\mathsf{R}) \in \mathcal{R} : \mathsf{R}^{\mathcal{I}_\mathcal{C}} = (\mathsf{R}^{\mathcal{I}_\mathcal{C}})^{+}$. If there exist $(\mathsf{a}^{\mathcal{I}_\mathcal{C}}, \mathsf{b}^{\mathcal{I}_\mathcal{C}}) \in R^{\mathcal{I}_\mathcal{C}}$ and $(\mathsf{b}^{\mathcal{I}_\mathcal{C}}, \mathsf{c}^{\mathcal{I}_\mathcal{C}}) \in R^{\mathcal{I}_\mathcal{C}}$ then case 6 in Definition 26 must have been applied for each tuple. But then, a chain of roles from a to c exists as well (possibly with gaps and blocking individuals) such that $(\mathsf{a}^{\mathcal{I}_\mathcal{C}}, \mathsf{c}^{\mathcal{I}_\mathcal{C}})$ is added to $\mathsf{R}^{\mathcal{I}_\mathcal{C}}$ as well.

In the following we prove that $\mathcal{I}_\mathcal{C}$ satisfies every assertion in $\mathcal{A}$.

For any $\mathsf{a} \neq \mathsf{b} \in \mathcal{A}$ or $(\mathsf{a}, \mathsf{b}) : \mathsf{R} \in \mathcal{A}$, $\mathcal{I}_\mathcal{C}$ satisfies them by definition.

For any $(\mathsf{a}, \mathsf{x}) : \mathsf{f} \in \mathcal{A}$, $\mathcal{I}_\mathcal{C}$ satisfies them by definition.

For any $(\mathsf{x}_1, \ldots, \mathsf{x}_\mathsf{n}) : \mathsf{P} \in \mathcal{A}$, $\mathcal{I}_\mathcal{C}$ satisfies them by definition. Since $\mathcal{A}$ is clash-free there exists a variable assignment such that the conjunction of all predicate assertions is satisfied. The variable assignment can be computed because the concrete domain is required to be admissible.

Next we consider assertions of the form $\mathsf{a} : \mathsf{C}$. We show by induction on the structure of C that $\mathsf{a}^{\mathcal{I}_\mathcal{C}} \in \mathsf{C}^{\mathcal{I}_\mathcal{C}}$.

If C is a concept name, then $\mathsf{a}^{\mathcal{I}_\mathcal{C}} \in \mathsf{C}^{\mathcal{I}_\mathcal{C}}$ by definition of $\mathcal{I}_\mathcal{C}$.

If $\mathsf{C} = \neg \mathsf{D}$, then D is a concept name since all concepts are in negation normal form (see Definition 15). $\mathcal{A}$ is clash-free and cannot contain $\mathsf{a} : \mathsf{D}$. Thus, $\mathsf{a} \notin \mathsf{D}^{\mathcal{I}_\mathcal{C}}$, i.e. $\mathsf{a}^{\mathcal{I}_\mathcal{C}} \in \Delta_{\mathcal{I}_\mathcal{C}} \setminus \mathsf{D}^{\mathcal{I}_\mathcal{C}}$. Hence $\mathsf{a}^{\mathcal{I}_\mathcal{C}} \in (\neg \mathsf{D})^{\mathcal{I}_\mathcal{C}}$.

If $\mathsf{C} = \mathsf{C}_1 \sqcap \mathsf{C}_2$ then (since $\mathcal{A}$ is complete) $\mathsf{a} : \mathsf{C}_1 \in \mathcal{A}$ and $\mathsf{a} : \mathsf{C}_2 \in \mathcal{A}$. By induction hypothesis, $\mathsf{a}^{\mathcal{I}_\mathcal{C}} \in {\mathsf{C}_1}^{\mathcal{I}_\mathcal{C}}$ and $\mathsf{a}^{\mathcal{I}_\mathcal{C}} \in {\mathsf{C}_2}^{\mathcal{I}_\mathcal{C}}$. Hence $\mathsf{a}^{\mathcal{I}_\mathcal{C}} \in (\mathsf{C}_1 \sqcap \mathsf{C}_2)^{\mathcal{I}_\mathcal{C}}$.

If $\mathsf{C} = \mathsf{C}_1 \sqcup \mathsf{C}_2$ then (since $\mathcal{A}$ is complete) either $\mathsf{a} : \mathsf{C}_1 \in \mathcal{A}$ or $\mathsf{a} : \mathsf{C}_2 \in \mathcal{A}$. By induction hypothesis, $\mathsf{a}^{\mathcal{I}_\mathcal{C}} \in {\mathsf{C}_1}^{\mathcal{I}_\mathcal{C}}$ or $\mathsf{a}^{\mathcal{I}_\mathcal{C}} \in {\mathsf{C}_2}^{\mathcal{I}_\mathcal{C}}$. Hence $\mathsf{a}^{\mathcal{I}_\mathcal{C}} \in (\mathsf{C}_1 \sqcup \mathsf{C}_2)^{\mathcal{I}_\mathcal{C}}$.

If $\mathsf{C} = \forall \mathsf{R} \,.\, \mathsf{D}$, then we have to show that for all $\mathsf{b}^{\mathcal{I}_\mathcal{C}}$ with $(\mathsf{a}^{\mathcal{I}_\mathcal{C}}, \mathsf{b}^{\mathcal{I}_\mathcal{C}}) \in \mathsf{R}^{\mathcal{I}_\mathcal{C}}$ it holds that $\mathsf{b}^{\mathcal{I}_\mathcal{C}} \in \mathsf{D}^{\mathcal{I}_\mathcal{C}}$. If $(\mathsf{a}^{\mathcal{I}_\mathcal{C}}, \mathsf{b}^{\mathcal{I}_\mathcal{C}}) \in \mathsf{R}^{\mathcal{I}_\mathcal{C}}$, then according to Definition 26, b is a successor of a via a chain of roles $\mathsf{S}_\mathsf{i} \in \mathsf{R}^{\downarrow}$ or there exists corresponding blocking individuals as domain element of $\mathsf{S}_\mathsf{i} \in \mathsf{R}^{\downarrow}$, i.e. the chain might contain "gaps" with associated blocking individuals (see Figure 1). Since $(\mathsf{a}^{\mathcal{I}_\mathcal{C}}, \mathsf{b}^{\mathcal{I}_\mathcal{C}}) \in \mathsf{R}^{\mathcal{I}_\mathcal{C}}$ and ${\mathsf{S}_\mathsf{i}}^{\mathcal{I}_\mathcal{C}} \subseteq \mathsf{R}^{\mathcal{I}_\mathcal{C}}$ there exists tuples $({\mathsf{c}_\mathsf{i}}^{\mathcal{I}_\mathcal{C}}, {\mathsf{c}_{\mathsf{i}+1}}^{\mathcal{I}_\mathcal{C}}) \in {\mathsf{S}_\mathsf{i}}^{\mathcal{I}_\mathcal{C}}$. Due to Definition 26 it holds that $\forall \mathsf{i} \in 1..\mathsf{n} : \exists \mathsf{R}' \in T, \mathsf{R}' \in \mathsf{R}^{\downarrow}, \mathsf{S}_\mathsf{i} \in \mathsf{R}'^{\downarrow}$. Therefore $\mathsf{c}_\mathsf{k} : \forall \mathsf{R}' \,.\, \mathsf{D} \in \mathcal{A}, (\mathsf{k} \in 1..\mathsf{n}-1)$ because $\mathcal{A}$ is complete. For the same reason $\mathsf{b} : \mathsf{D} \in \mathcal{A}$. By induction hypothesis it holds that $\mathsf{b}^{\mathcal{I}_\mathcal{C}} \in \mathsf{D}^{\mathcal{I}_\mathcal{C}}$. As mentioned before, the chain of roles can have one or more "gaps" (see Figure 1). However, due to Definition 26 in case of a "gap" there exists a blocking individual such that a similar argument as in case 6 can be applied, i.e. in case of a gap between c_i and $\mathsf{c}_{\mathsf{i}+1}$ with blocking individual d_i for c_i, the blocking condition ensures that the concept set of the blocking individual is a superset of the concept set of the blocked individual. Since it is assumed that $(\mathsf{d_i}, \mathsf{c_{i+1}}) : \mathsf{S_{i+1}} \in \mathcal{A}$ and $\mathcal{A}$ is complete it holds that $\mathsf{c_{i+1}} : \forall\, \mathsf{R'} \,.\, \mathsf{D} \in \mathcal{A}$. Applying the same agument inductively, we can conclude that $\mathsf{c_{n-1}} : \forall\, \mathsf{R'} \,.\, \mathsf{D} \in \mathcal{A}$ and again, we have $\mathsf{b}^{\mathcal{I_C}} \in \mathsf{D}^{\mathcal{I_C}}$ by induction hypothesis.

If $\mathsf{C} = \exists\, \mathsf{R} \,.\, \mathsf{D}$, then we have to show that there exists an individual $\mathsf{b}^{\mathcal{I_C}} \in \Delta_{\mathcal{I_C}}$ with $(\mathsf{a}^{\mathcal{I_C}}, \mathsf{b}^{\mathcal{I_C}}) \in \mathsf{R}^{\mathcal{I_C}}$ and $\mathsf{a}^{\mathcal{I_C}} \in \mathsf{D}^{\mathcal{I_C}}$. Since ABox $\mathcal{A}$ is complete, we have either $(\mathsf{a}, \mathsf{b}) : \mathsf{S} \in \mathcal{A}$ with $\mathsf{S} \in \mathsf{R}^{\downarrow}$ and $\mathsf{b} : \mathsf{D} \in \mathcal{A}$ or a is blocked by an individual c and $(\mathsf{c}, \mathsf{b}) : \mathsf{S} \in \mathcal{A}$ (again $\mathsf{S} \in \mathsf{R}^{\downarrow}$). In the first case we have $(\mathsf{a}^{\mathcal{I_C}}, \mathsf{b}^{\mathcal{I_C}}) \in \mathsf{R}^{\mathcal{I_C}}$ by the definition of $\mathcal{I_C}$ (case 6, $\mathsf{n} = 1, \mathsf{c_i} = \mathsf{d_i}$) and $\mathsf{b}^{\mathcal{I_C}} \in \mathsf{D}^{\mathcal{I_C}}$ by induction hypothesis. In the second case there exists the blocking individual c with $\mathsf{c} : \exists\, \mathsf{S} \,.\, \mathsf{D} \in \mathcal{A}$ and $\mathsf{S} \in \mathsf{R}^{\downarrow}$. By definition c cannot be blocked and by hypothesis $\mathcal{A}$ is complete. For a blocked individual there exists no direct successor. Hence, two individuals cannot block each other. This is due to the fact that for an individual b being a blocking individual for c the ordering $\mathsf{b} \prec \mathsf{c}$ must hold. However, no generating rule is applicable to a blocked individual and, therefore $\mathsf{b} \prec \mathsf{c}$ does not hold. So we have an individual b with $(\mathsf{c}, \mathsf{b}) : \mathsf{S} \in \mathcal{A}$ and $\mathsf{b} : \mathsf{D} \in \mathcal{A}$. By induction hypothesis we have $\mathsf{b}^{\mathcal{I_C}} \in \mathsf{D}^{\mathcal{I_C}}$ and by the definition of $\mathcal{I_C}$ (case 6, $\mathsf{n} = 1, \mathsf{c_i} \neq \mathsf{d_i}$, $\mathsf{d_i}$ is a blocking individual for $\mathsf{c_i}$ and $\mathsf{a} = \mathsf{c_i}, \mathsf{c} = \mathsf{d_i}$) we have $(\mathsf{a}^{\mathcal{I_C}}, \mathsf{b}^{\mathcal{I_C}}) \in \mathsf{R}^{\mathcal{I_C}}$.

If $\mathsf{C} = \exists_{\geq n}\, \mathsf{R}$, we prove the hypothesis by contradiction. We assume that $\mathsf{a}^{\mathcal{I_C}} \notin (\exists_{\geq n}\, \mathsf{R})^{\mathcal{I_C}}$. Then there exist at most m $(0 \leq m < n)$ distinct S-successors of a with $\mathsf{S} \in \mathsf{R}^{\downarrow}$. Two cases can occur: (1) the individual a is not blocked in $\mathcal{I_C}$. Then we have less than n S-successors of a in $\mathcal{A}$ and the R$\exists_{\geq n}$-rule is applicable to a. This contradicts the assumption that $\mathcal{A}$ is complete. (2) a is blocked by an individual c but the same argument as in case (1) holds and leads to the same contradiction.

For $\mathsf{C} = \exists_{\leq n}\, \mathsf{R}$ we show the goal by contradiction. Suppose that $\mathsf{a} \notin (\exists_{\leq n}\, \mathsf{R})^{\mathcal{I_C}}$. Then there exist at least $n+1$ distinct individuals ${\mathsf{b_1}}^{\mathcal{I_C}}, \ldots, {\mathsf{b}_{n+1}}^{\mathcal{I_C}}$ such that $(\mathsf{a}^{\mathcal{I_C}}, {\mathsf{b}_i}^{\mathcal{I_C}}) \in \mathsf{R}^{\mathcal{I_C}}$, $i \in 1..n+1$. The following two cases can occur. (1) The individual a is not blocked: We have $n+1$ $(\mathsf{a}, \mathsf{b}_i) : \mathsf{S}_i \in \mathcal{A}$ with $\mathsf{S}_i \in \mathsf{R}^{\downarrow}$ and $\mathsf{S}_i \notin T$, $i \in 1..n+1$. The R$\exists_{\leq n}$ rule cannot be applicable since $\mathcal{A}$ is complete and the b_i are distinct, i.e. $\mathsf{b}_i \not\doteq \mathsf{b}_j \in \mathcal{A}$, $i, j \in 1..n+1$, $i \neq j$. This contradicts the assumption that $\mathcal{A}$ is clash-free. (2) There exists a blocking individual c for a with $(\mathsf{c}, \mathsf{b}_i) : \mathsf{S}_i \in \mathcal{A}$, $\mathsf{S}_i \in \mathsf{R}^{\downarrow}$, and $\mathsf{S}_i \notin T$, $i \in 1..n+1$. This leads to an analogous contradiction. Due to the construction of the canonical interpretation in case of a blocking condition and a non-transitive role R (R is required to be a simple role, see the syntactic restrictions for number restrictions), there is no $(\mathsf{a}^{\mathcal{I_C}}, {\mathsf{b}_k}^{\mathcal{I_C}}) \in \mathsf{R}^{\mathcal{I_C}}$ if there is no $(\mathsf{c}^{\mathcal{I_C}}, {\mathsf{b}_k}^{\mathcal{I_C}}) \in \mathsf{R}^{\mathcal{I_C}}$.

If $\mathsf{C} = \exists\, \mathsf{f_1}, \ldots, \mathsf{f_n} \,.\, \mathsf{P}$ we show that there exist concrete objects $y_1, \ldots, y_n \in \Delta_{\mathcal{D}}$ such that $(\mathsf{a}^{\mathcal{I_C}}, y_1) \in {\mathsf{f_1}}^{\mathcal{I_C}}, \ldots, (\mathsf{a}^{\mathcal{I_C}}, y_n) \in {\mathsf{f_n}}^{\mathcal{I_C}}$ and $(y_1, \ldots, y_n) \in \mathsf{P}^{\mathcal{I_C}}$. The R$\exists$P rule generates assertions $(\mathsf{a}, \mathsf{x_1}) : \mathsf{f_1}, \ldots, (\mathsf{a}, \mathsf{x_n}) : \mathsf{f_n}, (\mathsf{x_1}, \ldots, \mathsf{x_n}) : \mathsf{P}$. Since $\mathcal{A}$ is clash-free there is no concrete domain clash. Hence there exists a variable assignment α that maps $\mathsf{x_1}, \ldots, \mathsf{x_n}$ to elements of $\Delta_\mathcal{D}$. The conjunction of concrete domain predicates is satisfiable and $(\mathsf{x_1}^{\mathcal{I_C}}, \ldots, \mathsf{x_n}^{\mathcal{I_C}}) \in \mathsf{P}^{\mathcal{I_C}}$. By definition of $\mathcal{I_C}$ it holds that $(\mathsf{a}^{\mathcal{I_C}}, \mathsf{x_1}^{\mathcal{I_C}}) \in \mathsf{f_1}^{\mathcal{I_C}}, \ldots, (\mathsf{a}^{\mathcal{I_C}}, \mathsf{x_n}^{\mathcal{I_C}}) \in \mathsf{f_n}^{\mathcal{I_C}}$. Thus, there exist $y_1, \ldots, y_n$ such that the above-mentioned requirements are fulfilled and therefore $\mathsf{a}^{\mathcal{I_C}} \in (\exists\, \mathsf{f_1}, \ldots, \mathsf{f_n} \,.\, \mathsf{P})^{\mathcal{I_C}}$

If $\mathsf{C} = \forall\, \mathsf{f} \,.\, \bot_\mathcal{D}$ then we show by contradiction that $\mathsf{a}^{\mathcal{I_C}} \in (\forall\, \mathsf{f} \,.\, \bot_\mathcal{D})^{\mathcal{I_C}}$. Because $\mathcal{A}$ is clash-free, there cannot be an assertion $(\mathsf{a}, \mathsf{x}) : \mathsf{f} \in \mathcal{A}$ for some x in O_c and an $\mathsf{f} \in F$. Thus, it does not hold that there exists $(\mathsf{a}^{\mathcal{I_C}}, y) \in \mathsf{f}^{\mathcal{I_C}}$ and hence $\mathsf{a}^{\mathcal{I_C}} \in (\forall\, \mathsf{f} \,.\, \bot_\mathcal{D})^{\mathcal{I_C}}$.

If $\forall\, x \,.\, x : \mathsf{D} \in \mathcal{A}$, then –due to the completeness of $\mathcal{A}$– for each individual a in $\mathcal{A}$ we have $\mathsf{a} : \mathsf{D} \in \mathcal{A}$ and, by the previous cases, $\mathsf{a}^{\mathcal{I_C}} \in \mathsf{D}^{\mathcal{I_C}}$. Thus, $\mathcal{I_C}$ satisfies $\forall\, x \,.\, x : \mathsf{D}$. Finally, since $\mathcal{I_C}$ satisfies all assertions in $\mathcal{A}$, $\mathcal{I_C}$ satisfies $\mathcal{A}$.

□

Theorem 28 (Completeness) Let $\mathcal{A}_\mathcal{T}$ be an augmented ABox and $\mathcal{R}$ be a role box, then there exists at least one completion $\mathcal{A}'$ being computed by applying the completion rules w.r.t. the role box $\mathcal{R}$.

Proof. By contraposition: Obviously, an Abox containing a clash is inconsistent. If every completion of $\mathcal{A}$ is inconsistent, then it follows from Proposition 21 that the ABox $\mathcal{A}$ is inconsistent w.r.t. the role box $\mathcal{R}$. □

Definition 29 Let $\mathcal{A}$ be a completion of an augmented ABox. Then, $||n_\mathcal{A} := \{\mathsf{C} | \exists\, (\mathsf{a} : \mathsf{D}) \in \mathcal{A} : \mathsf{C} \in \mathit{subs}(\mathsf{D}), \text{or}\ \exists (\forall\, x \,.\, x : \mathsf{D}) \in \mathcal{A} : \mathsf{C} \in \mathit{subs}(\mathsf{D})\}||$ is called the maximum number of concepts in $\mathcal{A}$. The function *subs* applied to a concept D returns the set of all concepts appearing as substrings in D (incl. D).

Note that $n_\mathcal{A}$ is bounded by the length of the string of the augmented ABox $\mathcal{A}$. In the following we assume that $|| \cdot ||$ returns the cardinality of a set plus 1.

Lemma 30 Let $\mathcal{A}$ be a completion of an augmented ABox $\mathcal{A}_\mathcal{T}$. Furthermore, let $T_\mathcal{R}$ be the finite set of transitive roles mentioned in a role box $\mathcal{R}$. In any set X consisting of individuals occurring in $\mathcal{A}$ with a cardinality greater than $2^{||T_\mathcal{R}|| \times n_\mathcal{A}}$ there exist at least two individuals $\mathsf{a}, \mathsf{b} \in X$ whose concept sets are equal ($\mathsf{a} \equiv_{\mathcal{A}'} \mathsf{b}$).

Proof. The only rule that generates assertions with concepts not already mentioned in $\mathcal{A}$ is the R$\forall_+$C rule. New concepts of the form $\forall\, \mathsf{T} \,.\, \mathsf{C}$ may be generated. The number of these concepts is bounded by $||T_\mathcal{R}|| \times n_\mathcal{A}$ because there are only $||T_\mathcal{R}||$ transitive roles mentioned in the role box $\mathcal{R}$ and only $n_\mathcal{A}$ different concepts in $\mathcal{A}$. There cannot exist more than $2^{||T_\mathcal{R}|| \times n_\mathcal{A}}$ different concept sets for the individuals in $\mathcal{A}'$. If we have $2^{||T_\mathcal{R}|| \times n_\mathcal{A}}$ individuals with different concept sets, then there can be no additional individual with a new concept set. □

Lemma 31 Let $\mathcal{A}_{\mathcal{T}}$ be an augmented ABox and let $\mathcal{A}'$ be a completion of $\mathcal{A}_{\mathcal{T}}$ w.r.t. $\mathcal{R}$. Furthermore, let $T_{\mathcal{R}}$ be the finite set of transitive roles mentioned in the role box $\mathcal{R}$. Then, there occur at most $2^{||T_{\mathcal{R}}|| \times n_{\mathcal{A}}}$ non-blocked new individuals in $\mathcal{A}'$.

Proof. Suppose we have $2^{||T_{\mathcal{R}}|| \times n_{\mathcal{A}}} + 1$ non-blocked new individuals in $\mathcal{A}'$. From Lemma 30 we know that there exist at least two individuals a, b in $\mathcal{A}'$ such that $\mathsf{a} \equiv_{\mathcal{A}'} \mathsf{b}$. By Definition 18 we have either $\mathsf{a} \prec \mathsf{b}$ or $\mathsf{b} \prec \mathsf{a}$. Assume without loss of generality that $\mathsf{a} \prec \mathsf{b}$ holds and $\mathsf{a} \equiv_{\mathcal{A}'} \mathsf{b}$ implies $\sigma(\mathcal{A}', \mathsf{a}) \supseteq \sigma(\mathcal{A}', \mathsf{b})$. Then we have either $\mathsf{a} \succcurlyeq_{\mathcal{A}'} \mathsf{b}$ or there exists an individual c with $\mathsf{c} \succcurlyeq_{\mathcal{A}'} \mathsf{b}$ and $\mathsf{c} \prec \mathsf{a}$. Both cases contradict the hypothesis. □

Theorem 32 (Termination) Let $\mathcal{A}_{\mathcal{T}}$ be an augmented ABox. Every completion of $\mathcal{A}_{\mathcal{T}}$ w.r.t. a role box $\mathcal{R}$ is finite and its size is $O(2^{4n})$ where $n = ||T_{\mathcal{R}}|| \times n_0$.

Proof. Let $\mathcal{A}'$ be a completion of $\mathcal{A}_{\mathcal{T}}$. From Lemma 31 we know that $\mathcal{A}'$ has at most $2^{||T|| \times n_{\mathcal{A}'}} \leq 2^n$ non-blocked new individuals. Therefore, a total of at most $m \times 2^n$ new individuals may exists in $\mathcal{A}'$, where m is the maximum number of direct successors for any individual in $\mathcal{A}'$.

Note that m is bounded by the number of $\exists\, \mathsf{R} \,.\, \mathsf{C}$ concepts ($\leq n$) plus the total sum of numbers occurring in $\exists_{\geq n} \mathsf{R}$. Since numbers are expressed in binary, their sum is bounded by $2^{n_0} (\leq 2^n)$. Hence, we have $m \leq 2^n + n$. Since the number of individuals in the initial ABox is also bounded by n, the total number of individuals in $\mathcal{A}'$ is at most $m \times (2^n + n) \leq (2^n + n) \times (2^n + n)$, i.e. $O(2^{2n})$.

The number of different assertions of the form $\mathsf{a} : \mathsf{C}$ or $\forall\, x \,.\, x : \mathsf{C}$ in which each individual in $\mathcal{A}'$ can be involved, is bounded by n and each assertion has a size linear in n. Hence, the total size of these assertions is bounded $n \times n \times 2^{2n}$, i.e. $O(2^{3n})$.

The number of different assertions of the form $(\mathsf{a}, \mathsf{b}) : \mathsf{R}$ or $\mathsf{a} \neq \mathsf{b}$ is bounded by $(2^{2n})^2$, i.e. $O(2^{4n})$.

The number of different assertions of the form $(\mathsf{a}, \mathsf{x}) : \mathsf{f}$ is bounded by $O(2^{2n})$ due to fork elimination.

The number of different assertions of the form $(\mathsf{x}_1, \ldots, \mathsf{x}_n) : \mathsf{P}$ is bounded by $n + (n \times 2^{2n})$, i.e. $O(2^{3n})$. The initial set of concrete domain predicate assertions is bounded by n. In addition, for each individual there may be n concept assertions yielding additional predicate assertions.

In conclusion, we have a size of $O(2^{4n})$ for $\mathcal{A}'$. □

Theorem 33 (Decidability) Checking whether a knowledge base $(\mathcal{T}, \mathcal{R}, \mathcal{A})$ is consistent is a decidable problem.

Proof. Given a knowledge base $(\mathcal{T}, \mathcal{R}, \mathcal{A})$, an augmented ABox $\mathcal{A}_{\mathcal{T}}$ can be constructed in linear time. Thus, the claim follows immediately from Lemma 16 and Theorems 27, 28, and 32. □

5 Applying $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$: Configuration Revisited

In the previous section the decidability of the ABox consistency problem for $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ has been shown. Thus, in principle all configuration problems formalized as knowledge bases in the language $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ as indicated in Section 3 can be solved. If the input knowlege base is consistent, the configuration will be represented by a model represented by the canonical interpretation derived from a completion that is computed by the algorithm discussed above. However, due to the fact that the algorithm is nondeterministic, some problems might remain.

5.1 Unintended Blocking

In the context of configuration, blocking might lead to an undesirable model. Let us consider the ABox $\{\mathsf{a}:\mathsf{A} \sqcup \mathsf{C}\}$, the TBox $\{\mathsf{C} \sqsubseteq \exists\, \mathsf{R} \,.\, \mathsf{A} \sqcup \mathsf{C}\}$ and an empty role box. One possible completion that might be derived by a concrete implementation of the knowledge base consistency algorithm is the following:

$\{\forall\, x \,.\, x : (\neg\mathsf{C} \sqcup \exists\, \mathsf{R} \,.\, \mathsf{A} \sqcup \mathsf{C}),$
$\mathsf{a} : (\mathsf{A} \sqcup \mathsf{C}), \mathsf{a} : (\neg\mathsf{C} \sqcup \exists\, \mathsf{R} \,.\, \mathsf{A} \sqcup \mathsf{C}), \mathsf{a} : \mathsf{C}, \mathsf{a} : \exists\, \mathsf{R} \,.\, \mathsf{A} \sqcup \mathsf{C},$
$(\mathsf{a}, \mathsf{b}) : \mathsf{R}, \mathsf{b} : (\mathsf{A} \sqcup \mathsf{C}), \mathsf{b} : (\neg\mathsf{C} \sqcup \exists\, \mathsf{R} \,.\, \mathsf{A} \sqcup \mathsf{C}), \mathsf{b} : \mathsf{C}, \mathsf{b} : \exists\, \mathsf{R} \,.\, \mathsf{A} \sqcup \mathsf{C},$
$(\mathsf{b}, \mathsf{c}) : \mathsf{R}, \mathsf{c} : (\mathsf{A} \sqcup \mathsf{C}), \mathsf{c} : (\neg\mathsf{C} \sqcup \exists\, \mathsf{R} \,.\, \mathsf{A} \sqcup \mathsf{C}), \mathsf{c} : \mathsf{C}, \mathsf{c} : \exists\, \mathsf{R} \,.\, \mathsf{A} \sqcup \mathsf{C}\}$

This is a completion with c being blocked. Hence, the canonical interpretation contains a loop w.r.t. to the role R. Whether this is acceptable or not might depend on the application context of the configuration solution. However, it should be noted that in this specific case there also exists a completion without a blocked individual. The configuration example presented in Section 3 is solved without blocking.

5.2 Limited Expressivity

For configuring a motor, a set of cylinders all of which have equal piston displacements might be required. However, with $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ concrete domains predicates can only be established for a single individual, i.e. a single cylinder, rather than between different cylinders. A whole set of cylinders being part of a motor can only be constrained using an ABox and respective concrete domain assertions. Thus, only a fixed set of individuals can be considered during the configuration process. If it is not clear in beforehand whether a 4-, 6- or 8-cylinder engine will be required, a more expressive description logic is needed.

5.3 Analysis of an Extension of $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$

A possibility for extending the expressivity of $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ might be to employ the predicate exists restriction of $\mathcal{ALC}(\mathcal{D})$ which offers feature chains [Baader and Hanschke, 1991a]. We call the language $\mathcal{ALCNH}_{R^+}(\mathcal{D})$. Unfortunately, it holds that $\mathcal{ALCNH}_{R^+}$ augmented with a predicate exists restriction supporting feature chains as in $\mathcal{ALC}(\mathcal{D})$ is undecidable. In [Lutz, 1999] it is shown that $\mathcal{ALC}(\mathcal{D})$ with generalized inclusion axioms (GCIs) is undecidable. $\mathcal{ALCNH}_{R^+}$ offers role hierarchies and transitive roles which provide the same expressivity as GCIs.

An undecidability proof may lead to insights about how to come up with new operators or syntactic restrictions of existing operators in order to develop a representation language that can cope with specific application requirements not covered by less expressive (decidable) languages. Since the GCI-based undecidability proof with Turing machines presented in [Lutz, 1999] is rather involved, we give a more direct proof based on transitive roles and role hierarchies and demonstrate that even if TBoxes are discarded, $\mathcal{ALCNH}_{R^+}$ with concrete domains is undecidable in general.

The syntax and semantics of $\mathcal{ALCNH}_{R^+}(\mathcal{D})$ is a slightly modified variant of $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$.

In Definition 1 for $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ the set of simple roles S is introduced. In $\mathcal{ALCNH}_{R^+}(\mathcal{D})$ a specific subset $A \subseteq S$ of simple roles called *attributes* is distinguished.

If $\mathsf{a_1 a_2 \cdots a_{n-1}}$ are attributes and $\mathsf{f_n}$ is a feature, then a composition of attributes and features (written $\mathsf{a_1 a_2 \cdots a_{n-1} f_n}$) is called a chain (with length n). A single feature (i.e. a chain of length 1) is also called a chain. If $\mathsf{P} \in \Phi_{\mathcal{D}}$ is a predicate of the concrete domain $\mathcal{D}$ and $\mathsf{u_1, \ldots, u_k}$ are chains, then the following expression is a concept term: $\exists\, \mathsf{u_1, \ldots, u_k} \,.\, \mathsf{P}$ *(predicate exists restriction)*. In addition to $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$, attributes can be used instead of roles in value and exists restrictions.

Each attribute a from A is mapped to a partial function $\mathsf{a}^{\mathcal{I}}$ from $\Delta_{\mathcal{I}}$ to $\Delta_{\mathcal{I}}$. If $\mathsf{u} = \mathsf{a_1 \cdots a_{n-1} f_n}$ is a chain, then $\mathsf{u}^{\mathcal{I}}$ denotes the composition $\mathsf{a_1} \circ \ldots \circ \mathsf{a_{n-1}} \circ \mathsf{f_n}$ of partial functions $\mathsf{a_1}^{\mathcal{I}}, \ldots, \mathsf{a_{n-1}}^{\mathcal{I}} \mathsf{f_n}^{\mathcal{I}}$. The interpretation function is modified as follows:

$$(\exists\, \mathsf{u_1}, \ldots, \mathsf{u_n} \,.\, \mathsf{P})^{\mathcal{I}} := \{a \in \Delta_{\mathcal{I}} \mid \exists x_1, \ldots, x_n \in \Delta_{\mathcal{D}} : (a, x_1) \in \mathsf{u_1}^{\mathcal{I}}, \ldots, (a, x_n) \in \mathsf{u_n}^{\mathcal{I}}, (x_1, \ldots, x_n) \in \mathsf{P}^{\mathcal{I}}\}$$

Proposition 34 (Undecidability of $\mathcal{ALCNH}_{R^+}(\mathcal{D})$) The concept consistency problem for $\mathcal{ALCNH}_{R^+}(\mathcal{D})$ is not decidable.

The proposition can be proven by a reduction from the Post Correspondence Problem (PCP). The general idea of the proof is a slight variation of the undecidability proofs for the description logics $\mathcal{ALC}(\mathcal{D})$ with a transitivity operator [Baader and Hanschke, 1992] and $\mathcal{ALCRP}(\mathcal{D})$ [Lutz, 1998; Lutz and Möller, 1997].

Proof. A Post Correspondence Problem S is defined as follows. Given a nonempty finite set $S = \{(l_i, r_i) \mid i = 1, \ldots m\}$, where l_i and r_i are words over an alphabet Σ, a solution of S is a sequence of indices $i_1, \ldots, i_k$ with $k \geq 1$ such that the concatenations $wl = l_{i_1} \ldots l_{i_k}$ and $wr = r_{i_1} \ldots l_{i_k}$ denote the same word. The PCP is known to be undecidable if Σ contains at least two symbols.

For the reduction, the elements of Σ are viewed as digits from $\{1, \ldots, \text{B-1}\}$ at base B, where $B := |\Sigma| + 1$. $\overline{w}$ denotes the nonnegative integer at base 10 which the (nonempty) word w represents at base B (see also [Baader and Hanschke, 1992]). If vw is the concatenation of two words $v, w \in \Sigma^*$, then $\overline{vw} = \overline{v} * B^{|w|} + \overline{w}$, where $|w|$ is the length of the word w. The function $w \mapsto \overline{w}$ is a 1–1-mapping from Σ^* into the set of nonnegative integers. Let w_l, w_r be features and $f_1, \ldots, f_m$ be attributes. Furthermore, let R be a transitive superrole of the attributes f_i $(i \in 1, \ldots, m)$. Then, for a given instance S of the PCP we define a concept $C(S)$:

$$\begin{array}{rcl} C(S) & \doteq & \exists w_l.\mathit{null\text{-}p} \sqcap \exists w_r.\mathit{null\text{-}p} \\ & & \sqcap_{i=1}^{m} (\exists w_l, f_i w_l.\mathit{constr\text{-}p}_l^i \sqcap \exists w_r, f_i w_r.\mathit{constr\text{-}p}_r^i) \sqcap \\ & & \forall R. \sqcap_{i=1}^{m} (\exists w_l, f_i w_l.\mathit{constr\text{-}p}_l^i \sqcap \exists w_r, f_i w_r.\mathit{constr\text{-}p}_r^i) \sqcap \\ & & \forall R.\exists w_l, w_r.\mathit{notequal\text{-}p} \end{array}$$

The predicates used are defined as follows:

$$\begin{array}{rcl} \mathit{null\text{-}p}(a) & := & a = 0 \\ \mathit{constr\text{-}p}_l^i(a, b) & := & b = \overline{l_i} + a * B^{|l_i|} \\ \mathit{constr\text{-}p}_r^i(a, b) & := & b = \overline{r_i} + a * B^{|r_i|} \\ \mathit{notequal\text{-}p}(a, b) & := & a \neq b \end{array}$$

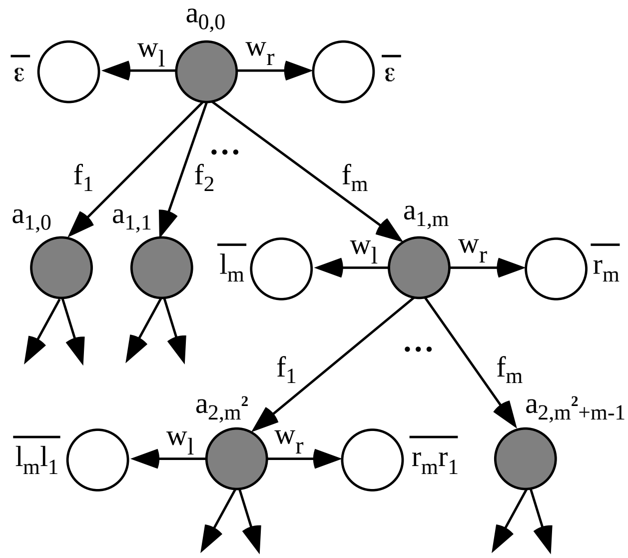

Figure 2: Search space of the Post Correspondence Problem encoded as a model of a concept $C(S)$.

The undecidability of $\mathcal{ALCNH}_{R^+}(\mathcal{D})$ is proven by showing that $C(S)$ is consistent iff the PCP S has no solution. Therefore, if the consistency of $C(S)$ could be decided, the algorithm could also be used to decide if a PCP S has a solution.

We first show that S has no solution if $C(S)$ is consistent. This can be easily seen by considering the definition of $C(S)$. If $C(S)$ is consistent there must exist an interpretation $\mathcal{I}$ with $C(S)^{\mathcal{I}} \neq \emptyset$. Figure 2 demonstrates that the interpretation encodes the (infinite) search space for a solution of S. However, since $C(S)$ is assumed to be consistent, $\forall R.\exists w_l, w_r.notequal\text{-}p$ holds. Therefore, none of the paths in the search space leads to a solution.

Now we prove that $C(S)$ is consistent if S has no solution. This direction is proven by defining an interpretation with $C(S)^{\mathcal{I}} \neq \emptyset$ for a PCP S for which it is known that no solution exists.

$$\Delta_{\mathcal{I}} = \{a_{ij} \,|\, i \geq 0, 1 \leq j < m^i\};$$
$$\forall i \geq 0, 0 \leq j < m^i :$$
$$f_1{}^{\mathcal{I}}(a_{ij}) = a_{i+1\ j*m}, \ \ldots, \ f_m{}^{\mathcal{I}}(a_{ij}) = a_{i+1\ j*m+m-1},$$
$$w_l{}^{\mathcal{I}}(a_{ij}) = \overline{\phi_l(i,j)}, \quad w_r{}^{\mathcal{I}}(a_{ij}) = \overline{\phi_r(i,j)}$$

where ϕ_l and ϕ_r are two recursively defined concatenation functions (*concat* concatenates words and $\lfloor\rfloor$ denotes the $floor$ function):

$$\begin{aligned}\phi_l(0,0) &= \epsilon\\ \phi_r(0,0) &= \epsilon\\ \phi_l(i,j) &= concat(\phi_l(i-1,\lfloor j/m \rfloor), l_{j+1-(m*\lfloor j/m \rfloor)})\\ \phi_r(i,j) &= concat(\phi_r(i-1,\lfloor j/m \rfloor), r_{j+1-(m*\lfloor j/m \rfloor)}).\end{aligned}$$

□

As we have discussed before, the undecidability proof for $\mathcal{ALCNH}_{R^+}(\mathcal{D})$ presented here follows the approach for showing the undecidability of $\mathcal{ALC}(\mathcal{D})$-trans presented in [Baader and Hanschke, 1992]. Furthermore, the idea to construct a concept $C(S)$ in such a way that it is satisfiable iff the PCP S has *no* solution has been taken from [Lutz and Möller, 1997; Lutz, 1998; Haarslev et al., 1998]. The basic idea of the undecidability proofs is to construct a transitive role in order to propagate a concept constraint to all individuals in the tree which encodes the search space of a PCP. In the undecidability proof for $\mathcal{ALCNH}_{R^+}(\mathcal{D})$ presented here, a similar effect is achieved by exploiting role hierarchies and transitive roles.[4]

Analyzing the model of the PCP it becomes clear that the undecidability is caused by the possibility to establish predicates for conrete domain objects that are referred to via features with different individuals on the left-hand side of the corresponding ABox assertions. The finite model property is lost in $\mathcal{ALCNH}_{R^+}(\mathcal{D})$. However, as long as a finite model is actually found by a calculus, this is no problem. So there might be some hope that "conditions" under which non-termination is "likely to occur" can be established. If these conditions are encountered, then the answer to the inference problem could be "unknown".

Rather than considering the quite complex PCP concept in detail, we discuss a simpler $\mathcal{ALCNH}_{R^+}(\mathcal{D})$ concept intended for describing lists of numbers.[5] As in the previous subsection, there are predicates established for concrete objects that are referred to by different individuals.

Let us assume, car is a feature cdr is an attribute and Rest is a transitive superrole of cdr. We also use the name cadr for the chain $\mathsf{cdr}\,\mathsf{car}$. Let P, $\mathsf{Q1}$ and $\mathsf{Q2}$ be elements of $\Phi_{\Re}$ (see above) such that $\mathsf{P(x,y)} := \mathsf{y} - \mathsf{x} = 1$, $\mathsf{Q1(x)} := \mathsf{x} > 7$ and $\mathsf{Q2(x)} := \mathsf{x} = 100$.

Example 1: $\exists\,\mathsf{car},\mathsf{cadr}\,.\,\mathsf{P} \sqcap \forall\,\mathsf{Rest}\,.\,(\exists\,\mathsf{car},\mathsf{cadr}\,.\,\mathsf{P})$

[4] Decidability problems with concrete domains and cyclic axioms are also discussed in [Buchheit et al., 1995].

[5] In a configuration context, for instance, a list of cylinders might be described.

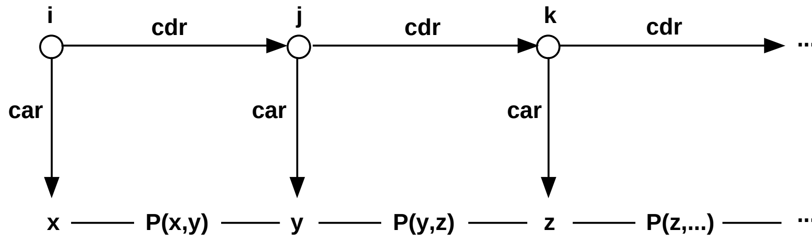

Figure 3: List of numbers greater than 7 decreasing by 1.

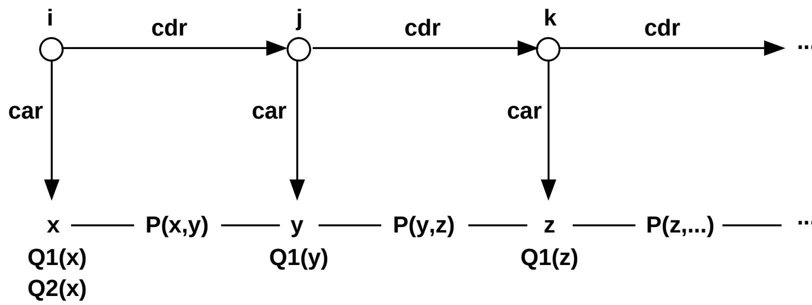

Figure 4: List of numbers greater than 7 decreasing by 1 starting at 100.

Figure 3 sketches a model for this concept (i, j and k are individuals and $\mathsf{x}, \mathsf{y}, \mathsf{z}$ are concrete objects). Since P is based on a total strict ordering, the model for the concept in Example 1 must be infinite.

Example 2: $\exists\,\mathsf{car}\,.\,\mathsf{Q1} \sqcap \exists\,\mathsf{car}\,.\,\mathsf{Q2}$
$\sqcap \exists\,\mathsf{car},\mathsf{cadr}\,.\,\mathsf{P} \sqcap \forall\,\mathsf{Rest}\,.\,(\exists\,\mathsf{car}\,.\,\mathsf{Q1} \sqcap \exists\,\mathsf{car},\mathsf{cadr}\,.\,\mathsf{P})$

Figure 4 shows an interpretation which is to be continued to the right in the expected way. Since x is equal to 100 it can easily be seen that this interpretation cannot be a model because it must be extended to the right until some 'successor' (filler of the role Rest) will be less than 7.

Example 3: $\forall\,\mathsf{cdr}\,.\,\bot \sqcup (\exists\,\mathsf{car},\mathsf{cadr}\,.\,\mathsf{P} \sqcap \forall\,\mathsf{Rest}\,.\,(\forall\,\mathsf{cdr}\,.\,\bot \sqcup \exists\,\mathsf{car},\mathsf{cadr}\,.\,\mathsf{P}))$

A model for this concept has the structure of the interpretation shown in Figure 3 but can be finite because there is no role filler for cdr required. Even the interpretation consisting only of one individual without fillers for cdr and car is a model. This interpretation represents an empty list.

From an application-oriented point of view, it is often not necessary to describe *infinite* lists. The concept in Example 3 captures that lists can be of *arbitrary but finite* length.[6] Since there exists a finite model it might be

[6]It should be emphasized that, obviously, the concept of Example 3 is by no means equisatisfiable compared to the concept of Example 2.

possible to devise a calculus to compute a configuration based on an initial input ABox (cf. Figure 4). However, since the language is undecidable in general, a sound and complete (and terminating) calculus for deciding knowledge base consistency inevitably must return 'unknown' in some situations. We conjecture that it might be possible to detect these situations (i.e. guarantee termination) while perserving that both 'yes' and 'no' answers can be trusted. In the case of "linear" structures as discussed with the examples above it might be possible to integrate additional proof techniques involving the induction principle. In Example 1 and Example 2, "unknown" might be returned. Details of a calculus still have to be worked out.

6 Conclusion

We presented a tableaux calculus deciding the knowledge base consistency problem for the description logic $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$. Applications of the logic in the context of configuration problems have been sketched. The Cylinder example demonstrates that some requirements of a model-based configuration system are fulfilled by $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$. The calculus presented in this paper can be used to solve "simple" configuration problems in which the configuration space can be described by an $\mathcal{ALCNH}_{R^+}(\mathcal{D})^-$ knowledge base (see [Cunis et al., 1991; Buchheit et al., 1995; Günter, 1995] for additional representation structures for solving configuration problems).

A highly optimized variant of the calculus for the sublogic $\mathcal{ALCNH}_{R^+}$ is already implemented in the ABox description logic system RACE [Haarslev et al., 1999].[7] RACE will be extended with support for reasoning with concrete domains in the near future. The adaption of important optimization techniques such as dependency-directed backtracking and model merging to concrete domains is discussed in [Turhan and Haarslev, 2000; Turhan, 2000] (see also [Haarslev and Möller, 2000c] for extended model merging algorithms).

Acknowledgments

We would like to thank Carsten Lutz and Ulrike Sattler for thoughtful comment on an earlier version of this report. All deficiencies are due to our own faults, of course.

[7] RACE is available at http://kogs-www.informatik.uni-hamburg.de/~race/

References

Baader, F. (1999). Logic-based knowledge representation. In Wooldridge, M. and Veloso, M., editors, *Artificial Intelligence Today, Recent Trends and Developments*, number 1600 in Lecture Notes in Computer Science, pages 13–41. Springer Verlag.

Baader, F. and Hanschke, P. (1991a). A scheme for integrating concrete domains into concept languages. In *Twelfth International Conference on Artificial Intelligence, Darling Harbour, Sydney, Australia, Aug. 24-30, 1991*, pages 452–457.

Baader, F. and Hanschke, P. (1991b). A scheme for integrating concrete domains into concept languages. Technical Report DFKI-RR-91-10, German Center for AI (DFKI).

Baader, F. and Hanschke, P. (1992). Extensions of concept languages for a mechanical engineering application. In Ohlbach, H., editor, *Proceedings, GWAI-92: Advances in Artificial Intelligence, 16th German Conference on Artificial Intelligence*, pages 132–143. Springer Verlag, Berlin.

Baader, F. and Sattler, U. (2000). Tableau algorithms for description logics. In Dyckhoff, R., editor, *Proceedings of the International Conference on Automated Reasoning with Tableaux and Related Methods (Tableaux 2000)*, volume 1847 of *Lecture Notes in Artificial Intelligence*, pages 1–18, St Andrews, Scotland, UK. Springer-Verlag.

Baader et al., F., editor (2000). *Proceedings of the International Workshop on Description Logics (DL'2000), August 17 - August 19, 2000, Aachen, Germany.*

Buchheit, M., Donini, F., and Schaerf, A. (1993). Decidable reasoning in terminological knowledge representation systems. *Journal of Artificial Intelligence Research*, 1:109–138.

Buchheit, M., Klein, R., and Nutt, W. (1994). Configuration as model construction: The constructive problem solving approach. In Sudweeks, F. and Gero, J., editors, *Proc. 4th International Conference on Artificial Intelligence in Design, Lausanne (Switzerland)*. Kluwer, Dordrecht.

Buchheit, M., Klein, R., and Nutt, W. (1995). Constructive problem solving: A model construction approach towards configuration. Technical Report DFKI-TM-95-01, German Center for AI (DFKI).

Cunis, R. (1991). Modellierung technischer Systeme in der Begriffshierarchie. In [Cunis et al., 1991], chapter 5.

Cunis, R., Günter, A., and Strecker, H., editors (1991). *Das PLAKON-Buch – Ein Expertensystemkern für Planungs- und Konfigurierungsaufgaben in technischen Domänen*, volume 266. Springer-Verlag.

Donini, F., Lenzerini, M., Nardi, D., and Schaerf, A. (1996). Reasoning in description logics. In Brewka, G., editor, *Principles of Knowledge Representation*. CSLI Publications.

Günter, A., editor (1995). *Wissensbasiertes Konfigurieren – Ergebnisse aus dem Projekt PROKON*. infix, Sankt Augustin.

Haarslev, V., Lutz, C., and Möller, R. (1998). Foundations of spatioterminological reasoning with description logics. In Cohn, T., Schubert, L., and Shapiro, S., editors, *Proceedings of Sixth International Conference on Principles of Knowledge Representation and Reasoning (KR'98), Trento, Italy, June 2-5, 1998*, pages 112–123.

Haarslev, V. and Möller, R. (2000a). Expressive ABox reasoning with number restrictions, role hierachies, and transitively closed roles. In Cohn, A., Giunchiglia, F., and Selman, B., editors, *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR'2000), Breckenridge, Colorado, USA, 2000*.

Haarslev, V. and Möller, R. (2000b). High performance reasoning with very large knowledge bases. In [Baader et al., 2000]. In print.

Haarslev, V. and Möller, R. (2000c). Optimizing TBox and ABox reasoning with pseudo models. In [Baader et al., 2000]. In print.

Haarslev, V., Möller, R., and Turhan, A.-Y. (1999). RACE user's guide and reference manual version 1.1. Technical Report FBI-HH-M-289/99, University of Hamburg, Computer Science Department. Available at URL http://kogs-www.informatik.uni-hamburg.de/~haarslev/publications/report-FBI-289-99.ps.gz.

Horrocks, I., Sattler, U., and Tobies, S. (1999). Practical reasoning for expressive description logics. In Ganzinger, H., McAllester, D., and Voronkov, A., editors, *Proceedings of the 6th International Conference on Logic for Programming and Automated Reasoning (LPAR'99)*, number 1705 in Lecture Notes in Artificial Intelligence, pages 161–180. Springer-Verlag.

Horrocks, I., Sattler, U., and Tobies, S. (2000). Reasoning with individuals for the description logic $\mathcal{SHIQ}$. In MacAllester, D., editor, *Proceedings of the 17th International Conference on Automated Deduction (CADE-17)*, Lecture Notes in Computer Science, Germany. Springer Verlag.

Lutz, C. (1998). Representation of Topological Information in Description Logics (in German). Master's thesis, University of Hamburg, Computer Science Department.

Lutz, C. (1999). The complexity of reasoning with concrete domains (revised version). LTCS-Report 99-01, LuFG Theoretical Computer Science, RWTH Aachen, Germany. See http://www-lti.informatik.rwth-aachen.de/Forschung/Papers.html.

Lutz, C. and Möller, R. (1997). Defined topological relations in description logics. In Rousset et al., M.-C., editor, *Proceedings of the International Workshop on Description Logics, DL'97, Sep. 27-29, 1997, Gif sur Yvette, France*, pages 15–19. Universite Paris-Sud, Paris.

Sattler, U. (1996). A concept language extended with different kinds of transitive roles. In Görz, G. and Hölldobler, S., editors, *20. Deutsche Jahrestagung für Künstliche Intelligenz*, number 1137 in Lecture Notes in Artificial Intelligence, pages 333–345. Springer Verlag, Berlin.

Schmidt-Schauss, M. and Smolka, G. (1991). Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26.

Schröder, C., Möller, R., and Lutz, C. (1996). A partial logical reconstruction of PLAKON/KONWERK. In Baader, F., Bürckert, H.-J., Günter, A., and Nutt, W., editors, *Proceedings of the Workshop on Knowledge Representation and Configuration WRKP'96*, number D-96-04 in DFKI-Memos, pages 55–64.

Turhan, A.-Y. (2000). Optimierungsmethoden für den Erfüllbarkeitstest bei Beschreibungslogiken mit konkreter Domäne (in German). Diploma Thesis (Diplomarbeit).

Turhan, A.-Y. and Haarslev, V. (2000). Adapting optimization techniques to description logics with concrete domains. In [Baader et al., 2000]. In print.

Woods, W. and Schmolze, J. (1992). The KL-ONE family. In Lehmann, F., editor, *Semantic Networks in Artificial Intelligence*, pages 133–177. Pergamon Press, Oxford.

Wright, J., Weixelbaum, E., Vesonder, G., Brown, K., Palmer, S., Berman, J., and Moore, H. (1993). A knowledge-based configurator that supports sales, engineering, and manufacturing at at&t network systems. *AI Magazine*, 14(3):69–80.