Uniform and Modular Sequent Systems for Description Logics

Tim Lyon[1], Jonas Karge[1]

[1]Computational Logic Group, Institute of Artificial Intelligence, Technische Universität Dresden, 01062 Dresden, Germany

Abstract

We introduce a framework that allows for the construction of sequent systems for expressive description logics extending $\mathcal{ALC}$. Our framework not only covers a wide array of common description logics, but also allows for sequent systems to be obtained for extensions of description logics with special formulae that we call *role relational axioms*. All sequent systems are sound, complete, and possess favorable properties such as height-preserving admissibility of common structural rules and height-preserving invertibility of rules.

Keywords

Sequent Calculus, Description Logics, Proof theory

1. Introduction

Description logics (DLs) consist of an assortment of knowledge representation languages used to structure and represent knowledge in an unequivocal and perspicuous manner. In DLs, knowledge is represented by means of *knowledge bases (KBs)*, i.e. collections of expressions involving concepts and roles. KBs contain explicit knowledge of a particular domain of interest, and by means of logical consequence, implicit knowledge may be derived, thus giving rise to a need for logical tools to extract information. In addition, it is reasonable to request that such tools be *automatable*, i.e. it is not only desirable to develop tools that have the potential of deriving information, but which give definitive answers to a problem by means of an algorithm. It is also worthwhile to possess tools that allow one to constructively prove (meta-)logical properties of DLs (e.g. concept interpolation, or re-writings of concepts and TBoxes), and which are applicable to a wide array of DLs, regardless of their idiosyncrasies.

Such tools—meeting the above demands—are capable of being developed on the basis of proof theory. Indeed, various DLs have been equipped with tableau-based proof-search algorithms [1, 2, 3, 4, 5, 6], resolution-based algorithms [7, 8, 9], or consequence-based algorithms [10, 11], to solve certain reasoning tasks. These works highlight and demonstrate the success of proof-theoretic methods in application to problems of description logics. Therefore, a proof-theoretic formalism that yields proof systems for a significant number of DLs *on demand* is desirable.

DL 2022: 35th International Workshop on Description Logics, August 7–10, 2022, Haifa, Israel
timothy_stephen.lyon@tu-dresden.de (T. Lyon); jonas.karge@tu-dresden.de (J. Karge)
https://iccl.inf.tu-dresden.de/web/Tim_Lyon (T. Lyon); https://iccl.inf.tu-dresden.de/web/Jonas_Karge (J. Karge)
0000-0003-3214-0828 (T. Lyon); 0000-0002-8296-1010 (J. Karge)

Hence, the intent of this paper is to propose a uniform and modular framework for generating proof systems—namely, *sequent systems*—for a large class of DLs, in the style of [12]. That is, the purpose of this paper is to provide a general recipe for constructing sequent systems for DLs.

Although work has been done on supplying sequent systems for DLs [13, 14, 15, 16], the systems have been constructed for a relatively narrow set. The distinguishing feature of the present paper is that we provide a formalism for generating sound and complete sequent systems for a sizable class of expressive DLs. Indeed, our work not only covers $\mathcal{ALC}$ and its prominent extensions (e.g. $\mathcal{SHIOQ}$ and the DL $\mathcal{SROIQ}$ that underlies OWL 2 [17]), but allows for extensions of expressive DLs with axioms we refer to as *role relational axioms (RRAs)*. Such axioms express properties of, and relationships between, roles. For instance, $\mathsf{Trans}(r)$ and $\mathsf{Dis}(r, s)$, which express that the role r is transitive and the roles r and s are disjoint, respectively, are defined to be instances of role relational axioms. It will be seen that the sequent formalism we provide is both uniform, covering many DLs, and modular, meaning that a sequent system for one DL is straightforwardly transformable into a sequent system for another DL by the addition or deletion of inference rules. Due to space constraints we leave the discussion of complexity related issues as well as proof-search algorithms up to future work.

The paper is organized as follows: In (Section 2), we introduce expressive DLs, including their semantics and features of their knowledge bases. In (Section 3), we introduce a sequent calculus for the *attributive concept language with complements* $\mathcal{ALC}$ [6], and define extensions for other expressive DLs along with the addition of rules for RRAs. We argue that all of our sequent calculi are sound, complete, and possess standard properties (e.g. invertibility of rules and admissibility of contraction).

2. Description Logics

In this section, we present the family of expressive description logics (DLs) (cf. [18]) that will be considered in this paper. This class of logics is obtained by extending $\mathcal{ALC}$. We first define $\mathcal{ALC}$ and its associated semantics, and then discuss extensions thereof.

2.1. Preliminaries and $\mathcal{ALC}$

$\mathcal{ALC}$, and DLs more generally, are defined relative to a *vocabulary* $\mathcal{V} = (\mathbf{R}, \mathbf{C}, \mathbf{I})$ the components of which are taken to be pairwise disjoint, countable sets. Each set contains primitive symbols dedicated to a particular purpose: the set $\mathbf{R}$ contains *role names* used to denote binary relations, the set $\mathbf{C}$ contains *concept names* used to denote classes of entities, and the set $\mathbf{I}$ contains *individuals* used to denote particular entities. We use r, s, ... (potentially annotated) to denote role names, C, D, ... (potentially annotated) to denote concept names, and a, b, ... (potentially annotated) to denote individuals. For $\mathcal{ALC}$, *complex concepts* are built from role and concept names as dictated by the following BNF grammar:

$$P ::= C \mid \bot \mid \top \mid \neg P \mid P \sqcup P \mid P \sqcap P \mid \exists r.P \mid \forall r.P$$

where $C \in \mathbf{C}$ and $r \in \mathbf{R}$. We use the symbols P, Q, ... (potentially annotated) to denote complex concepts. We interpret complex concepts and roles as follows:

Definition 1 (Interpretation [1]). *An* interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *contains a non-empty set* $\Delta^{\mathcal{I}}$*, called the* domain*, and a map* $\cdot^{\mathcal{I}}$ *such that for every* $C \in \mathbf{C}$*,* $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$*; for every* $r \in \mathbf{R}$*,* $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$*; and for every* $a \in \mathbf{I}$*,* $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$*. The map* $\cdot^{\mathcal{I}}$ *is extended to complex concept names as follows:*

$\top^{\mathcal{I}} := \Delta^{\mathcal{I}}$; $\bot^{\mathcal{I}} := \emptyset$; $C \sqcup D^{\mathcal{I}} := C^{\mathcal{I}} \cup D^{\mathcal{I}}$; $C \sqcap D^{\mathcal{I}} := C^{\mathcal{I}} \cap D^{\mathcal{I}}$;
$\exists r.C^{\mathcal{I}} := \{a \in \Delta^{\mathcal{I}} \mid$ *there exists* $b \in \Delta^{\mathcal{I}}$ *s.t.* $(a, b) \in r^{\mathcal{I}}$ *and* $b \in C^{\mathcal{I}}$*.*$\}$;
$\forall r.C^{\mathcal{I}} := \{a \in \Delta^{\mathcal{I}} \mid$ *for each* $b \in \Delta^{\mathcal{I}}$*, if* $(a, b) \in r^{\mathcal{I}}$*, then* $b \in C^{\mathcal{I}}$*.*$\}$.

As is standard for DLs, we collect specific formulae into *TBoxes* to specify certain properties of, and relationships between, concepts and roles. For $\mathcal{ALC}$, a TBox is a finite set of *general concept inclusions (GCIs)*, which are formulae of the form $P \sqsubseteq Q$, where P and Q are complex concepts. As explained in the following section (Section 2.2), we allow for a larger variety of formulae in TBoxes for DLs more expressive than $\mathcal{ALC}$.

Typically, for DLs, assertional knowledge is represented by formulae that state whether or not an individual or pair of individuals participate in a concept or role. Such formulae, which are referred to as *assertions*, comprise the *ABox*. For $\mathcal{ALC}$, the ABox contains a finite number of *concept assertions* of the form $a : P$ (with P a complex concept and $a \in \mathbf{I}$) and a finite number of *role assertions* of the form $r(a, b)$ (with $r \in \mathbf{R}$ and $a, b \in \mathbf{I}$). A *knowledge base (KB)* $\mathcal{K}$ is defined to be a pair consisting of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$, i.e. $\mathcal{K} = (\mathcal{T}, \mathcal{A})$. Let us now define how interpretations can be extended to the formulae of TBoxes, ABoxes, and therefore, to KBs.

Definition 2 (Model [1]). *An interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ satisfies *a GCI* $P \sqsubseteq Q$*, written* $\mathcal{I} \models P \sqsubseteq Q$*, iff* $P^{\mathcal{I}} \subseteq Q^{\mathcal{I}}$*; a concept assertion* $a : P$*, written* $\mathcal{I} \models a : P$*, iff* $a^{\mathcal{I}} \in P^{\mathcal{I}}$*; and a role assertion* $r(a, b)$*, written* $\mathcal{I} \models r(a, b)$*, iff* $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$*. We say that an intepretation* $\mathcal{I}$ *is a* model *of a TBox* $\mathcal{T}$ *(ABox* $\mathcal{A}$*) iff it satisfies all formulae in* $\mathcal{T}$ *(all formulae in* $\mathcal{A}$*, resp.). An interpretation* $\mathcal{I}$ *is a* model *of a KB* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ *iff it is a model of* $\mathcal{T}$ *and* $\mathcal{A}$*.*

2.2. Extensions of $\mathcal{ALC}$

The sequent systems provided in the subsequent section allow for a sizable number of DLs to be captured proof-theoretically. We focus our attention on presenting well-known extensions of $\mathcal{ALC}$, making use of the well-established naming convention for DLs to do so. Also, we define how new formulae within extensions are satisfied by a given interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$.

$\underline{\mathcal{S}}$ Prepending the name of a DL with $\mathcal{S}$ (rather than $\mathcal{ALC}$) indicates that a TBox is permitted to include *transitivity axioms* of the form $\mathsf{Trans}(r)$, or equivalently, axioms of the form $r \circ r \sqsubseteq r$, where the *composition operation* $\circ$ is interpreted accordingly (with $r, s \in \mathbf{R}$): $(r \circ s)^{\mathcal{I}} :=$

$$\{(a, b) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid \text{there exists a } c \in \Delta^{\mathcal{I}} \text{ s.t. } (a, c) \in r^{\mathcal{I}} \text{ and } (c, b) \in s^{\mathcal{I}}.\}$$

$\mathcal{I}$ *satisfies* $\mathsf{Trans}(r)$, written $\mathcal{I} \models \mathsf{Trans}(r)$, *iff* $r^{\mathcal{I}}$ is transitive.[1]

$\underline{\mathcal{H}}$ Including an $\mathcal{H}$ in the name of a DL (e.g. $\mathcal{ALCH}$) indicates that *simple role inclusions axioms (RIAs)* of the form $r \sqsubseteq s$ with $r, s \in \mathbf{R}$ may be included in a TBox. $\mathcal{I}$ *satisfies* $r \sqsubseteq s$, written $\mathcal{I} \models r \sqsubseteq s$, *iff* $r^{\mathcal{I}} \subseteq s^{\mathcal{I}}$.

[1] $r^{\mathcal{I}}$ is transitive *iff* for all $a, b, c \in \Delta^{\mathcal{I}}$, if $(a, b), (b, c) \in r^{\mathcal{I}}$, then $(a, c) \in r^{\mathcal{I}}$.

$\underline{\mathcal{SR}}$ The most notable feature of DLs whose names are prepended with $\mathcal{SR}$ is that such logics allow for *complex role inclusion axioms (CRIAs)* of the form $r_1 \circ \cdots \circ r_n \sqsubseteq r$ to be included in a TBox.[2] Additionally, DLs from the $\mathcal{SR}$ family may include *reflexivity axioms* of the form $\mathsf{Refl}(r)$, *irreflexivity axioms* of the form $\mathsf{Irr}(r)$, *asymmetry axioms* of the form $\mathsf{Asy}(r)$, or *disjointness axioms* of the form $\mathsf{Dis}(r, s)$.[3]

- $\mathcal{I}$ *satisfies* $r_1 \circ \cdots \circ r_n \sqsubseteq r$, written $\mathcal{I} \models r_1 \circ \cdots \circ r_n \sqsubseteq r$, *iff* ${r_1}^{\mathcal{I}} \circ \cdots \circ {r_n}^{\mathcal{I}} \subseteq r^{\mathcal{I}}$;
- $\mathcal{I}$ *satisfies* $\mathsf{Refl}(r)$, written $\mathcal{I} \models \mathsf{Refl}(r)$, *iff* $r^{\mathcal{I}}$ is reflexive;
- $\mathcal{I}$ *satisfies* $\mathsf{Irr}(r)$, written $\mathcal{I} \models \mathsf{Irr}(r)$, *iff* $r^{\mathcal{I}}$ is irreflexive;
- $\mathcal{I}$ *satisfies* $\mathsf{Asy}(r)$, written $\mathcal{I} \models \mathsf{Asy}(r)$, *iff* $r^{\mathcal{I}}$ is asymmetric;
- $\mathcal{I}$ *satisfies* $\mathsf{Dis}(r, s)$, written $\mathcal{I} \models \mathsf{Dis}(r, s)$, *iff* $r^{\mathcal{I}}$ and $s^{\mathcal{I}}$ are disjoint.

$\underline{\mathcal{O}}$ Including an $\mathcal{O}$ in the name of a DL indicates that the set $\mathbf{C}$ of concept names includes *nominals* of the form $\{a\}$, for each $a \in \mathbf{I}$. We interpret nominals accordingly: $\{a\}^{\mathcal{I}} := \{a^{\mathcal{I}}\}$.

$\underline{\mathcal{I}}$ Including an $\mathcal{I}$ in the name of a DL indicates that the set $\mathbf{R}$ includes *inverse roles* of the form r^{-}, for each $r \in \mathbf{R}$. We interpret inverse roles accordingly: $r^{-\mathcal{I}} := \{(b, a) \mid (a, b) \in r^{\mathcal{I}}\}$.

$\underline{\mathcal{F}}$ An $\mathcal{F}$ in the name of a DL indicates that a TBox may include *functionality axioms* of the form $\mathsf{Funct}(r)$ for $r \in \mathbf{R}$. $\mathcal{I}$ *satisfies* $\mathsf{Funct}(r)$, written $\mathcal{I} \models \mathsf{Funct}(r)$, *iff* $r^{\mathcal{I}}$ is functional.[4]

$\underline{\mathcal{N}}$ The symbol $\mathcal{N}$ is included in the name of a DL when it includes *unqualified number restrictions* of the form $(\leqslant nr.\top)$ or $(\geqslant nr.\top)$ with $r \in \mathbf{R}$ among its concepts. We interpret unqualified number restrictions as follows:[5] $(\leqslant nr.\top)^{\mathcal{I}} := \{a \in \Delta^{\mathcal{I}} \mid \#\{b \mid (a, b) \in r^{\mathcal{I}}\} \leq n\}$ and $(\geqslant nr.\top)^{\mathcal{I}} := \{a \in \Delta^{\mathcal{I}} \mid \#\{b \mid (a, b) \in r^{\mathcal{I}}\} \geq n\}$.

$\underline{\mathcal{Q}}$ We use $\mathcal{Q}$ to indicate that a DL includes *qualified number restrictions* of the form $(\leqslant nr.P)$ or $(\geqslant nr.P)$ with $r \in \mathbf{R}$ among its concepts. We interpret qualified number restrictions accordingly: $(\leqslant nr.P)^{\mathcal{I}} := \{a \in \Delta^{\mathcal{I}} \mid \#\{b \mid (a, b) \in r^{\mathcal{I}} \text{ and } b : P\} \leq n\}$ and $(\geqslant nr.P)^{\mathcal{I}} := \{a \in \Delta^{\mathcal{I}} \mid \#\{b \mid (a, b) \in r^{\mathcal{I}} \text{ and } b : P\} \geq n\}$.

<u>Other Extensions</u> We may also extend $\mathcal{ALC}$ by permitting the inclusion of *equality* or *inequality* axioms of the form $a \approx b$ and $a \not\approx b$ (resp.) in a TBox, by permitting *negated role assertions* of the form $\neg r(a, b)$ in an ABox, by allowing for the *universal role* U to be included in $\mathbf{R}$ (interpreted $\mathsf{U}^{\mathcal{I}} := \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$), or by allowing the complex concept $\exists r.\mathsf{Self}$ for $r \in \mathbf{R}$ (interpreted $(\exists r.\mathsf{Self})^{\mathcal{I}} := \{a \mid (a, a) \in r^{\mathcal{I}}\}$). The semantics of (in)equalities and negated role assertions is as follows:

- $\mathcal{I}$ *satisfies* $a \approx b$, written $\mathcal{I} \models a \approx b$, *iff* $a^{\mathcal{I}} = b^{\mathcal{I}}$;
- $\mathcal{I}$ *satisfies* $a \not\approx b$, written $\mathcal{I} \models a \not\approx b$, *iff* $a^{\mathcal{I}} \neq b^{\mathcal{I}}$;
- $\mathcal{I}$ *satisfies* $\neg r(a, b)$, written $\mathcal{I} \models \neg r(a, b)$, *iff* $(a, b) \notin r^{\mathcal{I}}$.

---

[2] We note that syntactic conditions are usually imposed on the form of CRIAs in order to ensure the decidability of the resulting DL (e.g., see [1, 3]).

[3] Each property is defined as follows: (i) $r^{\mathcal{I}}$ is reflexive *iff* for each $a \in \Delta^{\mathcal{I}}$, $(a, a) \in r^{\mathcal{I}}$, (ii) $r^{\mathcal{I}}$ is irreflexive *iff* for each $a \in \Delta^{\mathcal{I}}$, $(a, a) \notin r^{\mathcal{I}}$, (iii) $r^{\mathcal{I}}$ is asymmetric *iff* for each $a, b \in \Delta^{\mathcal{I}}$, if $(a, b) \in r^{\mathcal{I}}$, then $(b, a) \notin r^{\mathcal{I}}$, and (iv) $r^{\mathcal{I}}$ and $s^{\mathcal{I}}$ are disjoint *iff* $r^{\mathcal{I}} \cap s^{\mathcal{I}} = \emptyset$.

[4] $r^{\mathcal{I}}$ is functional *iff* for all $a, b, c \in \Delta^{\mathcal{I}}$, if $(a, b), (a, c) \in r^{\mathcal{I}}$, then $b = c$.

[5] We use $\#S$ for a set S to denote the *cardinality* of the set.

3. Sequent Systems

Our proof systems consist of inference rules that manipulate sequents of the form $\Lambda := \mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}$, where $\mathcal{R}, \Sigma$ is referred to as the *antecedent* and $\Pi, \mathcal{Q}$ is referred to as the *consequent*. Note that Σ, Π, $\mathcal{R}$, and $\mathcal{Q}$ are taken to be (potentially empty) multisets of DL formulae. Σ and Π are multisets of formulae of the form $a : P$, called *internal formulae (IFs)*, where a ranges over the set of individuals $\mathbf{I}$, and P is a complex concept generated via the following grammar in BNF:

$$P ::= C \mid \bot \mid \top \mid \neg P \mid P \sqcup P \mid P \sqcap P \mid \exists r.P \mid \forall r.P \mid \{a\} \mid (\leqslant nr.P) \mid (\geqslant nr.P) \mid \exists r.\mathsf{Self}$$

with $C \in \mathbf{C}$, $r \in \mathbf{R}$ (which is potentially an inverse role s^- or the universal role U), $a \in \mathbf{I}$, and $n \in \mathbb{N}$. $\mathcal{R}$ and $\mathcal{Q}$ consist of formulae generated via the following grammar in BNF, and are referred to as *external formulae (EFs)*.

$$F ::= P \sqsubseteq Q \mid r(a,b) \mid \neg r(a,b) \mid \mathsf{Rel}(r_1, \ldots, r_n) \mid r_1 \circ \cdots \circ r_n \sqsubseteq r \mid a \approx b \mid a \not\approx b$$

where P and Q are complex concepts, $a, b \in \mathbf{I}$, $r_1, \ldots, r_n, r \in \mathbf{R}$ (and are potentially inverse roles s^- or the universal role U), and for each arity $n \in \mathbb{N}$, the relation name Rel ranges over a countable set of n-ary relation names. We note that transitivity axioms $\mathsf{Trans}(r)$, reflexivity axioms $\mathsf{Refl}(r)$, irreflexivity axioms $\mathsf{Irr}(r)$, asymmetry axioms $\mathsf{Asy}(r)$, disjointness axioms $\mathsf{Dis}(r,s)$, and functionality axioms $\mathsf{Funct}(r)$ are all instances of formulae of the form $\mathsf{Rel}(r_1, \ldots, r_n)$, which we refer to as *role relational axioms (RRAs)*. We use $F, G, \ldots$ to denote EFs defined by the grammar above. We distinguish EFs from IFs as EFs are those formulae which *govern reasoning with complex concepts*, i.e. of reasoning with IFs.

When supplying a calculus for a particular DL, we assume that the EFs and IFs occurring within sequents are restricted to those formulae allowed by the DL language under consideration. For example, for $\mathcal{ALC}$, we omit the inclusion of nominals, (un)qualified number restrictions, and $\exists r.\mathsf{Self}$ from occurring in IFs since such concepts are not included in $\mathcal{ALC}$'s language.

3.1. The System G3$\mathcal{ALC}$ and Descriptive Definitional Rules

We now present our calculus G3$\mathcal{ALC}$ for the DL $\mathcal{ALC}$ as well as define extensions of the calculus with *descriptive definitional rules (DDRs)*.[6] DDRs introduce RRAs into either the antecedent or consequent of a sequent, and thus provide our calculus with the capacity to handle such formulae. We discuss DDRs in detail below, and mention the DDRs that introduce widely-used RRAs such as transitivity axioms and reflexivity axioms. The calculus G3$\mathcal{ALC}$ is obtained by transforming the semantics of $\mathcal{ALC}$ into inference rules (cf. [12, 20, 21]), and is displayed in Figure 1. Note that in the $(id_{\mathbf{R}})$ rule we stipulate that F must be of the form $r(a,b)$ or $a \approx b$. We refer to the *principal formulae* of a rule as those formulae which are explicitly presented in the conclusion (e.g. $a : P \sqcup Q$ is the principal formula of $(\sqcup_l)$), and to the multisets $\mathcal{R}$, Σ, Π, and $\mathcal{Q}$ as *contexts*. Furthermore, we note that proofs/derivations are constructed by successively applying inference rules to *initial rules/sequents*, i.e. rules without premises (e.g. $(id_{\mathbf{C}})$, $(id_{\mathbf{R}})$, $(\bot_l)$, and $(\top_r)$), and the *height* of a proof is defined to be the longest sequence of sequents from the conclusion of the proof to an initial rule (cf. [12]).

[6] For a discussion of G3-style calculi, along with the G1 and G2 variants, see [19, Section 80].

$$\frac{}{\mathcal{R}, \Sigma, a : C \vdash a : C, \Pi, \mathcal{Q}}\ (id_{\mathbf{C}}) \qquad \frac{}{\mathcal{R}, \Sigma, F \vdash F, \Pi, \mathcal{Q}}\ (id_{\mathbf{R}})$$

$$\frac{}{\mathcal{R}, \Sigma, a : \bot \vdash \Pi, \mathcal{Q}}\ (\bot_l) \qquad \frac{\mathcal{R}, \Sigma \vdash a : \bot, \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}}\ (\bot_r) \qquad \frac{\mathcal{R}, \Sigma, a : \top \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}}\ (\top_l)$$

$$\frac{}{\mathcal{R}, \Sigma \vdash a : \top, \Pi, \mathcal{Q}}\ (\top_r) \qquad \frac{\mathcal{R}, \Sigma \vdash a : P, \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma, a : \neg P \vdash \Pi, \mathcal{Q}}\ (\neg_l) \qquad \frac{\mathcal{R}, \Sigma, a : P \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash a : \neg P, \Pi, \mathcal{Q}}\ (\neg_r)$$

$$\frac{\mathcal{R}, \Sigma, a : P \vdash \Pi, \mathcal{Q} \qquad \mathcal{R}, \Sigma, a : Q \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma, a : P \sqcup Q \vdash \Pi, \mathcal{Q}}\ (\sqcup_l) \qquad \frac{\mathcal{R}, \Sigma \vdash a : P, a : Q, \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash a : P \sqcup Q, \Pi, \mathcal{Q}}\ (\sqcup_r)$$

$$\frac{\mathcal{R}, \Sigma, a : P, a : Q \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma, a : P \sqcap Q \vdash \Pi, \mathcal{Q}}\ (\sqcap_l) \qquad \frac{\mathcal{R}, \Sigma \vdash a : P, \Pi, \mathcal{Q} \qquad \mathcal{R}, \Sigma \vdash a : Q, \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash a : P \sqcap Q, \Pi, \mathcal{Q}}\ (\sqcap_r)$$

$$\frac{\mathcal{R}, P \sqsubseteq Q, a : P, a : Q, \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, P \sqsubseteq Q, a : P, \Sigma \vdash \Pi, \mathcal{Q}}\ (\sqsubseteq_l) \qquad \frac{\mathcal{R}, \Sigma, b : P \vdash b : Q, \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash P \sqsubseteq Q, \Pi, \mathcal{Q}}\ (\sqsubseteq_r)^{\dagger}$$

$$\frac{\mathcal{R}, \Sigma, r(a,b), b : P \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma, a : \exists r.P \vdash \Pi, \mathcal{Q}}\ (\exists_l)^{\dagger} \qquad \frac{\mathcal{R}, \Sigma, r(a,b) \vdash a : \exists r.P, b : P, \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma, r(a,b) \vdash a : \exists r.P, \Pi, \mathcal{Q}}\ (\exists_r)$$

$$\frac{\mathcal{R}, \Sigma, r(a,b), a : \forall r.P, b : P \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma, r(a,b), a : \forall r.P \vdash \Pi, \mathcal{Q}}\ (\forall_l) \qquad \frac{\mathcal{R}, \Sigma, r(a,b) \vdash b : P, \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash a : \forall r.P, \Pi, \mathcal{Q}}\ (\forall_r)^{\dagger}$$

Figure 1: G3$\mathcal{ALC}$. † stipulates that the rule can be applied only if b is an eigenvariable, i.e. b does not occur in the conclusion of the rule.

DDRs are rules which are equivalent to, and obtained from, *descriptive definitions*. Descriptive definitions define properties of, and relationships between, roles; i.e. they define the necessary and sufficient conditions for which an RRA obtains. For instance, the formula $\mathsf{Trans}(r) \leftrightarrow \forall a \forall b \forall c (r(a,b) \wedge r(b,c) \rightarrow r(a,c))$ defines the RRA $\mathsf{Trans}(r)$ for the role r.

Definition 3 (Descriptive Definition)**.** *A* descriptive definition *is a formula of the form:*

$$\mathsf{Rel}(r_1, \ldots, r_l) \leftrightarrow \forall \vec{a}(F_1 \wedge \cdots \wedge F_n \rightarrow G_1 \vee \cdots \vee G_k)$$

such that each F_i and G_j is an EF of the form $r(a,b)$ or $a \approx b$, the individuals $\vec{a} := a_1, \ldots, a_m$ occur within $F_1 \wedge \cdots \wedge F_n$ (which is $\top$ if the conjunction is empty) and $G_1 \vee \cdots \vee G_k$ (which is $\bot$ if the disjunction is empty), and where the definiens (to the right of the bi-conditional) only makes reference to the roles r_1, …, r_l and/or equalities of the form $a \approx b$ (for a and b in $\vec{a}$).

Each descriptive definition of the above form can be transformed into a pair of left and right introduction rules (introducing the RRA $\mathsf{Rel}(r_1, \ldots, r_l)$) as shown below:

$$\frac{\left\{\mathcal{R}, \mathsf{Rel}(r_1, \ldots, r_l), \overline{F}, G_j, \Sigma \vdash \Pi, \mathcal{Q} \mid 1 \leq j \leq k\right\}}{\mathcal{R}, \mathsf{Rel}(r_1, \ldots, r_l), \overline{F}, \Sigma \vdash \Pi, \mathcal{Q}}\ (\mathsf{Rel}_l)$$

$$\frac{\mathcal{R}, \overline{F}, \Sigma \vdash \Pi, \overline{G}, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, \mathsf{Rel}(r_1, \ldots, r_l), \mathcal{Q}}\ (\mathsf{Rel}_r)^{\dagger}$$

We let $\overline{F} := F_1, \ldots, F_n$, $\overline{G} := G_1, \ldots, G_k$ and the side condition † states that (Rel_r) is applicable only if the individuals $\vec{a}$ (the collection of all individuals occurring within $\overline{F}$ and $\overline{G}$) are eigenvariables. (NB. Eigenvariables are individuals that do not occur in the conclusion of a rule, i.e. they are fresh in the premise(s), which ensures the soundness of rule applications; for a discussion on eigenvariables, see [12].) We let $\mathsf{G3}\mathcal{ALC}^{\star}$ denote $\mathsf{G3}\mathcal{ALC}$ extended with any finite number of DDR pairs $\{(\mathsf{Rel}_l), (\mathsf{Rel}_r)\}$, and note that such extensions give calculi for extensions of $\mathcal{ALC}$. For example, if we aim to provide a calculus for the DL $\mathcal{S}$, then our calculus must be capable of reasoning with transitivity axioms i.e. formulae of the form $\mathsf{Trans}(r)$ with $r \in \mathbf{R}$. $\mathsf{Trans}(r)$ can be defined by means of a descriptive definition, implying that we can obtain a calculus for the DL $\mathcal{S}$ by extending $\mathsf{G3}\mathcal{ALC}$ with the two rules shown below. (NB. The side condition † states that a, b, and c must be eigenvariables.)

$$\frac{\mathcal{R}, \mathsf{Trans}(r), r(a,b), r(b,c), r(a,c), \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \mathsf{Trans}(r), r(a,b), r(b,c), \Sigma \vdash \Pi, \mathcal{Q}}\ (\mathsf{Trans}(r)_l)$$

$$\frac{\mathcal{R}, r(a,b), r(b,c), \Sigma \vdash \Pi, r(a,c), \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \mathsf{Trans}(r), \Pi, \mathcal{Q}}\ (\mathsf{Trans}(r)_r)^{\dagger}$$

Some care must be taken when extending $\mathsf{G3}\mathcal{ALC}$ with DDRs. It is possible that certain properties of $\mathsf{G3}\mathcal{ALC}$, such as contraction hp-admissibility (see Theorem 2), are not immediately preserved in extensions of the calculus with DDRs. We apply a solution that is motivated by the work of [12]; namely, we can avoid such undesirable circumstances by ensuring that any extension of $\mathsf{G3}\mathcal{ALC}$ with DDRs adheres to the *closure condition*. (NB. For the remainder of the paper, we assume that every extension of $\mathsf{G3}\mathcal{ALC}$ satisfies the closure condition.)

Definition 4 (Closure Condition [12]). *A calculus with DDRs satisfies the* closure condition *iff for any DDR in the calculus which has a substitution instance containing duplicate principal formulae, the calculus also contains an instance of the rule with the duplicate formulae contracted.*

Since only a finite number of substitution instances produce duplicate principal formulae in a DDR, the closure condition will only add a finite number of rules in any extension of $\mathsf{G3}\mathcal{ALC}$.

We now define a semantics for sequents as this will be used for soundness and completeness.

Definition 5 (Sequent Semantics). *Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation. A sequent $\Lambda := \mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}$ is* satisfied *in $\mathcal{I}$, written $\mathcal{I} \models \Lambda$, iff if $\mathcal{I}$ satisfies all formulae in $\mathcal{R}, \Sigma$, then $\mathcal{I}$ satisfies some formula in $\mathcal{Q}, \Pi$. A sequent Λ is* falsified *in $\mathcal{I}$ iff $\mathcal{I} \not\models \Lambda$, i.e. Λ is not satisfied in $\mathcal{I}$. A sequent Λ is* valid*, written $\models \Lambda$, iff it is satisfiable in every interpretation, and is invalid otherwise.*

3.2. Rules for Extensions of $\mathcal{ALC}$

We discuss extensions of $\mathsf{G3}\mathcal{ALC}^{\star}$ with rules for deriving new concept assertions (e.g. unqualified number restrictions and nominals) and EFs (e.g. equalities and RIAs). We introduce these additional rules in the same manner as we introduced extensions of $\mathcal{ALC}$ in Section 2.2.

$\underline{\mathcal{S}}$ If the language of our DL includes role compositions, then the rules $(\circ_l)$ and $(\circ_r)$ (shown below) should be included in the corresponding calculus to allow reasoning with role compositions. (NB. s is permitted to be a chain $r_1 \circ \cdots \circ r_n$ of role compositions.) Since we can use axioms of the form $r \circ r \sqsubseteq r$ or $\mathsf{Trans}(r)$ to indicate that a role r is transitive, there are two distinct sets of rules which can be included in a calculus to allow reasoning with transitive roles.

First, if our DL allows for axioms of the form $r \circ r \sqsubseteq r$, then the composition rules, and restricted versions of the $(cria_l)$ and $(cria_r)$ rules (introduced in the $\mathcal{SR}$ subsection below) that only allow principal formulae of the form $r \circ r \sqsubseteq r$, should be included in the corresponding calculus. (NB. The side condition † on the $(\circ_l)$ rule stipulates that b is an eigenvariable.)

$$\frac{\mathcal{R}, r(a,b), s(b,c), \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, (r \circ s)(a,c), \Sigma \vdash \Pi, \mathcal{Q}} \ (\circ_l)^\dagger$$

$$\frac{\mathcal{R}, \Sigma \vdash \Pi, (r \circ s)(a,c), r(a,b), \mathcal{Q} \qquad \mathcal{R}, \Sigma \vdash \Pi, (r \circ s)(a,c), s(b,c), \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, (r \circ s)(a,c), \mathcal{Q}} \ (\circ_r)$$

Second, if we make use of transitivity axioms of the form $\mathsf{Trans}(r)$ in our DL, then the DDRs $(\mathsf{Trans}(r)_l)$ and $(\mathsf{Trans}(r)_r)$, introduced in the previous section, should be included in our calculus to ensure sound and complete reasoning with such formulae.

$\underline{\mathcal{H}}$ If we wish to enable reasoning with RIAs of the form $r \sqsubseteq s$ (e.g. as in $\mathcal{ALCH}$), then one should add restricted versions of the $(cria_l)$ and $(cria_r)$ rules (introduced in the $\mathcal{SR}$ subsection below) where $n = 1$, to ensure sound and complete reasoning with RIAs.

$\underline{\mathcal{SR}}$ To enable reasoning with CRIAs, the composition rules $(\circ_l)$ and $(\circ_r)$ should be included along with the following $(cria_l)$ and $(cria_r)$ rules. (NB. The side condition † on the $(cria_r)$ rule states that a and b must be eigenvariables. For readability, let F denote $r_1 \circ \cdots \circ r_n \sqsubseteq r$.)

$$\frac{\mathcal{R}, F, \Sigma \vdash \Pi, (r_1 \circ \cdots \circ r_n)(a,b), \mathcal{Q} \qquad \mathcal{R}, r(a,b), F, \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, F, \Sigma \vdash \Pi, \mathcal{Q}} \ (cria_l)$$

$$\frac{\mathcal{R}, (r_1 \circ \cdots \circ r_n)(a,b), \Sigma \vdash \Pi, r(a,b), \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, F, \mathcal{Q}} \ (cria_r)^\dagger$$

The (ir)reflexivity, asymmetry, and disjointness axioms can all be defined by means of descriptive definitions: $\mathsf{Refl}(r) \leftrightarrow \forall a(\top \rightarrow r(a,a))$, $\mathsf{Asy}(r) \leftrightarrow \forall a \forall b(r(a,b) \land r(b,a) \rightarrow \bot)$, $\mathsf{Irr}(r) \leftrightarrow \forall a(r(a,a) \rightarrow \bot)$, and $\mathsf{Dis}(r,s) \leftrightarrow \forall a \forall b(r(a,b) \land s(a,b) \rightarrow \bot)$. Thus, extending $\mathsf{G3}\mathcal{ALC}^\star$ with the corresponding DDRs provides our calculus with the capacity to reason with such axioms. All such DDRs can be obtained from the (Rel_l) and (Rel_r) rule schemata.

$\underline{\mathcal{O}}$ To enable reasoning with nominals, one should include the following rules along with the equality rules of the final subsection below.

$$\frac{\mathcal{R}, a \approx b, a : \{b\}, \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, a : \{b\}, \Sigma \vdash \Pi, \mathcal{Q}} \ (\{b\}^1_l) \qquad \frac{\mathcal{R}, \Sigma \vdash \Pi, a : \{b\}, a \approx b, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, a : \{b\}, \mathcal{Q}} \ (\{b\}^1_r)$$

$$\frac{\mathcal{R}, b : \{b\}, \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}} \ (\{b\}^2_l) \qquad \frac{}{\mathcal{R}, \Sigma \vdash \Pi, b : \{b\}, \mathcal{Q}} \ (\{b\}^2_r)$$

$\underline{\mathcal{I}}$ To add support for reasoning with inverse roles, one should not only allow inverse roles to appear in the relevant rules of the calculus (e.g. $(id_{\mathbf{R}})$, $(\exists_l)$, and $(\forall_r)$), but should also include the following two rules that encode the fact that the roles r and r^- are inverses.

$$\frac{\mathcal{R}, r(a,b), r^{-}(b,a), \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, r(a,b), \Sigma \vdash \Pi, \mathcal{Q}}\,(inv(r)_l) \qquad \frac{\mathcal{R}, r^{-}(a,b), r(b,a), \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, r^{-}(a,b), \Sigma \vdash \Pi, \mathcal{Q}}\,(inv(r^{-})_l)$$

$$\frac{\mathcal{R}, \Sigma \vdash \Pi, r(a,b), r^{-}(b,a), \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, r(a,b), \mathcal{Q}}\,(inv(r)_r) \qquad \frac{\mathcal{R}, \Sigma \vdash \Pi, r^{-}(a,b), r(b,a), \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, r^{-}(a,b), \mathcal{Q}}\,(inv(r^{-})_r)$$

$\underline{\mathcal{F}}$ Functionality axioms of the form $\mathsf{Funct}(r)$ can be defined by means of descriptive definitions; e.g. $\mathsf{Funct}(r) \leftrightarrow \forall a \forall b \forall c (r(a,b) \land r(a,c) \rightarrow b \approx c)$. We can make use of the (Rel_l) and (Rel_r) rule schemata to define DDRs for $\mathsf{Funct}(r)$. Hence, a calculus can be enabled to reason about functionality axioms by including the equality rules (introduced in final subsection below) along with the pair of DDRs obtained from the above descriptive defintion.

$\underline{\mathcal{N}}$ To allow reasoning with unqualified number restrictions, one makes use of versions of the $(\leqslant nr.P_l)$, $(\leqslant nr.P_r)$, $(\geqslant nr.P_l)$, and $(\geqslant nr.P_r)$ rules (shown in the next subsection $\mathcal{Q}$) where the first set of premises is omitted, and where the $b_i : P$ formulae are omitted from the remaining premises. We refer to each of these versions as $(\leqslant nr_l)$, $(\leqslant nr_r)$, $(\geqslant nr_l)$, and $(\geqslant nr_r)$, respectively. Additionally, the equality rules of the final subsection below should be included to ensure proper reasoning with equalities.

$\underline{\mathcal{Q}}$ To enable a calculus to derive theorems concerning qualified number restrictions, we add the following four rules along with the equality rules of the final subsection below. (NB. In the $(\leqslant nr.P_r)$ rule, $\dagger_1$ states that $b_0, \ldots, b_n$ must be eigenvariables and $\mathcal{Q}' := \{b_i \approx b_j \mid 0 \leq i < j \leq n\}$, and in the $(\geqslant nr.P_l)$ rule, $\dagger_2$ states that $b_1, \ldots, b_n$ must be eigenvariables and $\mathcal{Q}' := \{b_i \approx b_j \mid 1 \leq i < j \leq n\}$.)

$$\frac{\begin{array}{c}\Big\{\mathcal{R}, r(a,b_0), \ldots, r(a,b_n), \Sigma, a : (\leqslant nr.P) \vdash b_i : P, \Pi, \mathcal{Q} \mid 0 \leq i \leq n\Big\} \cup \\ \Big\{\mathcal{R}, b_i \approx b_j, r(a,b_0), \ldots, r(a,b_n), \Sigma, a : (\leqslant nr.P) \vdash \Pi, \mathcal{Q} \mid 0 \leq i < j \leq n\Big\}\end{array}}{\mathcal{R}, r(a,b_0), \ldots, r(a,b_n), \Sigma, a : (\leqslant nr.P) \vdash \Pi, \mathcal{Q}}\,(\leqslant nr.P_l)$$

$$\frac{\mathcal{R}, r(a,b_0), \ldots, r(a,b_n), \Sigma, b_0 : P, \ldots, b_n : P \vdash \Pi, \mathcal{Q}', \mathcal{Q}}{\mathcal{R}, \Sigma \vdash a : (\leqslant nr.P), \Pi, \mathcal{Q}}\,(\leqslant nr.P_r)^{\dagger_1}$$

$$\frac{\mathcal{R}, r(a,b_1), \ldots, r(a,b_n), \Sigma, b_1 : P, \ldots, b_n : P \vdash \Pi, \mathcal{Q}', \mathcal{Q}}{\mathcal{R}, \Sigma, a : (\geqslant nr.P) \vdash \Pi, \mathcal{Q}}\,(\geqslant nr.P_l)^{\dagger_2}$$

$$\frac{\begin{array}{c}\Big\{\mathcal{R}, r(a,b_1), \ldots, r(a,b_n), \Sigma \vdash b_i : P, a : (\geqslant nr.P), \Pi, \mathcal{Q} \mid 1 \leq i \leq n\Big\} \cup \\ \Big\{\mathcal{R}, b_i \approx b_j, r(a,b_1), \ldots, r(a,b_n), \Sigma \vdash a : (\geqslant nr.P), \Pi, \mathcal{Q} \mid 0 \leq i < j \leq n\Big\}\end{array}}{\mathcal{R}, r(a,b_1), \ldots, r(a,b_n), \Sigma \vdash a : (\geqslant nr.P), \Pi, \mathcal{Q}}\,(\geqslant nr.P_r)$$

Other Extensions To enable reasoning with equalities, we include $(\approx_l)$, $(\approx_r)$, $(\mathsf{Rep}_1(\approx))$, $(\mathsf{Rep}_2(\approx))$ and $(\mathsf{Euc}(\approx))$; to enable reasoning with inequalities, we add the $(\not\approx_l)$ and $(\not\approx_r)$ rules along with the previous five. To enable reasoning with negated role assertions we include $(\neg\mathbf{R}_l)$

and $(\neg \mathbf{R}_r)$ in our calculus; to ensure theorems can be derived concerning the universal role U, we allow the role to be used in the relevant rules of our calculus (e.g. $(id_{\mathbf{R}})$, $(\exists_l)$, and $(\forall_r)$) and also include the (U_l) and (U_r) rules shown below. Last, we include the (Self_l) and (Self_r) rules if we want our calculus to support complex concepts of the form $\exists r.\mathsf{Self}$. (NB. In the $(\mathsf{Rep}_1(\approx))$ and $(\mathsf{Rep}_2(\approx))$ rules, $[a/b]$ denotes a substitution of b for a in the relevant formula.)

$$\frac{\mathcal{R}, a \approx a, \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}} (\approx_l) \quad \frac{}{\mathcal{R}, \Sigma \vdash \Pi, a \approx a, \mathcal{Q}} (\approx_r) \quad \frac{\mathcal{R}, \Sigma \vdash \Pi, r(a,b), \mathcal{Q}}{\mathcal{R}, \neg r(a,b), \Sigma \vdash \Pi, \mathcal{Q}} (\neg \mathbf{R}_l)$$

$$\frac{\mathcal{R}, a \approx b, \Sigma, a : P, b : P \vdash \Pi, \mathcal{Q}}{\mathcal{R}, a \approx b, \Sigma, a : P \vdash \Pi, \mathcal{Q}} (\mathsf{Rep}_1(\approx)) \quad \frac{\mathcal{R}, \Sigma \vdash \Pi, r(a,a), \mathcal{Q}}{\mathcal{R}, \Sigma \vdash a : \exists r.\mathsf{Self}, \Pi, \mathcal{Q}} (\mathsf{Self}_r)$$

$$\frac{\mathcal{R}, a \approx b, F, F[a/b], \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, a \approx b, F, \Sigma \vdash \Pi, \mathcal{Q}} (\mathsf{Rep}_2(\approx)) \quad \frac{\mathcal{R}, a \approx b, a \approx c, b \approx c, \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, a \approx b, a \approx c, \Sigma \vdash \Pi, \mathcal{Q}} (\mathsf{Euc}(\approx))$$

$$\frac{\mathcal{R}, \Sigma \vdash \Pi, a \approx b, \mathcal{Q}}{\mathcal{R}, a \not\approx b, \Sigma \vdash \Pi, \mathcal{Q}} (\not\approx_l) \quad \frac{\mathcal{R}, r(a,b), \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, \neg r(a,b), \mathcal{Q}} (\neg \mathbf{R}_r) \quad \frac{\mathcal{R}, \mathsf{U}(a,b), \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}} (\mathsf{U}_l)$$

$$\frac{}{\mathcal{R}, \Sigma \vdash \Pi, \mathsf{U}(a,b), \mathcal{Q}} (\mathsf{U}_r) \quad \frac{\mathcal{R}, r(a,a), \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma, a : \exists r.\mathsf{Self} \vdash \Pi, \mathcal{Q}} (\mathsf{Self}_l) \quad \frac{\mathcal{R}, a \approx b, \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, a \not\approx b, \mathcal{Q}} (\not\approx_r)$$

We use $\mathsf{G3}\mathcal{ALC}^*$ to denote an extension of a calculus $\mathsf{G3}\mathcal{ALC}^\star$ with sets of the above rules. We allow for extensions with the sets shown below, and note that the addition of one set of rules may necessitate the addition of another set of rules, as explained above. Extensions with rules for RRAs (such as $\mathsf{Trans}(r)$ and $\mathsf{Asy}(r)$) are taken into account as extensions with DDRs:

$\{(\circ_l), (\circ_r)\}$; $\{(cria_l), (cria_r)\}$; $\{(\mathsf{Self}_l), (\mathsf{Self}_r)\}$; $\{(\not\approx_l), (\not\approx_r)\}$; $\{(\neg \mathbf{R}_l), (\neg \mathbf{R}_r)\}$;
$\{(\mathsf{U}_l), (\mathsf{U}_r)\}$; $\{(inv(r)_l), (inv(r^-)_l), (inv(r)_r), (inv(r^-)_r)\}$;
$\{(\{b\}^1_l), (\{b\}^2_l), (\{b\}^1_r), (\{b\}^2_r)\}$; $\{(\leqslant nr.P_l), (\leqslant nr.P_r), (\geqslant nr.P_l), (\geqslant nr.P_r)\}$;
$\{(\leqslant nr_l), (\leqslant nr_r), (\geqslant nr_l), (\geqslant nr_r)\}$; $\{(\approx_l), (\approx_r), (\mathsf{Euc}(\approx)), (\mathsf{Rep}_1(\approx)), (\mathsf{Rep}_2(\approx))\}$.

Theorem 1. *$\mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}$ is derivable in $\mathsf{G3}\mathcal{ALC}^*$ iff $\models \mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}$.*

Proof. Soundness (the forward direction) is shown by induction on the height of the given derivation. Completeness (the backward direction) is shown by a method due to Kripke [22]. We assume $\mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}$ is not derivable, and show that a counter-model can be extracted from failed proof search; thus, if a sequent is not derivable, it is not valid, implying completeness. □

We additionally show that our calculi possess desirable proof-theoretic properties. Before stating our theorem concerning which properties are possessed, we recall the definition of each property for the reader. A rule is defined to be (*height-preserving*) *admissible* in a calculus *iff* if the premise(s) of the rule is (are) derivable in the calculus (with a certain height), then the conclusion is derivable in the calculus (with a height less than or equal to the height of the premise(s)). Let us define the *inverse* of (R), written $(\hat{R})$, to be the rule obtained by switching the conclusion and the premise(s) of (R). A rule (R) is defined to be (*height-preserving*) *invertible* in a calculus *iff* $(\hat{R})$ is (height-presevering) admissible. That is, if there exists a derivation for

$$\frac{\mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \mathcal{R}', \Sigma, \Sigma' \vdash \Pi, \mathcal{Q}}\ (wk_l) \qquad \frac{\mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi, \Pi', \mathcal{Q}, \mathcal{Q}'}\ (wk_r) \qquad \frac{\mathcal{R}, \mathcal{R}', \mathcal{R}', \Sigma, \Sigma', \Sigma' \vdash \Pi, \mathcal{Q}}{\mathcal{R}, \mathcal{R}', \Sigma, \Sigma' \vdash \Pi, \mathcal{Q}}\ (ctr_l)$$

$$\frac{\mathcal{R}, \Sigma \vdash \Pi', \Pi', \Pi, \mathcal{Q}', \mathcal{Q}', \mathcal{Q}}{\mathcal{R}, \Sigma \vdash \Pi', \Pi, \mathcal{Q}', \mathcal{Q}}\ (ctr_r) \qquad \frac{\mathcal{R}, \Sigma \vdash \Pi, \mathcal{Q}}{(\mathcal{R}, \Sigma)[b/a] \vdash (\Pi, \mathcal{Q})[b/a]}\ (sub)$$

Figure 2: Admissible structural rules.

the conclusion, its premises can be derived as well [12]. As is common in the literature, we usually write *hp-admissible* and *hp-invertible* instead of *height-preserving admissible* and *height-preserving invertible*, and we remark that such properties are important as they can be leveraged to prove decidability of logics [19], to permit automated counter-model extraction [23], or to prove cut-elimination [12], among other applications. Note that in (sub), applying a substitution $[b/a]$ to a multiset is defined in the usual way as the replacement of all occurrences of a by b in the multiset. Last, we note that special (hp-)admissible structural rules are shown in Figure 2.

Theorem 2. *Each calculus* G3$\mathcal{ALC}^*$ *possesses the following properties: (i) For all EFs and IFs* X*,* $\mathcal{R}, X, \Sigma \vdash \Pi, X, \mathcal{Q}$ *is derivable in* G3$\mathcal{ALC}^*$*, (ii) All rules of* G3$\mathcal{ALC}^*$ *are hp-invertible, (iii) The* (sub)*,* (wk_l)*,* (wk_r)*,* (ctr_l)*, and* (ctr_r) *rules are hp-admissible in* G3$\mathcal{ALC}^*$*.*

Proof. (i) is shown by induction on the structure of X, and (ii) and (iii) are shown by induction on the height of the given derivation. □

4. Conclusion and Future Work

This paper provides a uniform framework for generating sequent systems on demand for a considerable number of expressive description logics including extensions with role relational axioms. All calculi are sound, complete, and possess standard properties. In future work, we aim to optimize our calculi by (i) simplifying the systems through confirming the admissibility of rules (e.g. $(\bot_r)$ and $(\top_l)$), (ii) applying a methodology called *structural refinement* [24], which has been used to ready proof systems for use in automated reasoning tasks [25, 23], and (iii) extending our formalism to a broader set of DLs (e.g. intuitionistic or constructive DLs [26, 27, 28]) which can be defined proof-theoretically.

We note that efficient reasoners, based on tableaux, for expressive DLs do already exist (e.g. HermiT [29]). However, since the current paper merely provides a *framework* for constructing sequent systems for expressive DLs, comparing decision algorithms based on our sequent systems with those based on existing tableaux must be left to future work. Nevertheless, sequent calculi have proven beneficial in establishing meta-logical properties, and thus, we aim to adapt existing methods for sequent systems to obtain constructive proofs of (various forms of) interpolation (as in [25, 30]), and to utilize our systems in computing re-writings of concepts and TBoxes. Last, we conjecture that cut-elimination holds for G3$\mathcal{ALC}$ when we restrict cuts to IFs, though we aim to investigate various forms of cut-elimination for all of our sequent calculi.

References

[1] F. Baader, I. Horrocks, C. Lutz, U. Sattler, Introduction to Description Logic, Cambridge University Press, 2017.

[2] F. M. Donini, M. Lenzerini, D. Nardi, W. Nutt, The complexity of concept languages, Information and Computation 134 (1997) 1–58. doi:10.1006/inco.1997.2625.

[3] I. Horrocks, U. Sattler, Decidability of shiq with complex role inclusion axioms, Artificial Intelligence 160 (2004) 79–104.

[4] M. Ortiz, D. Calvanese, T. Eiter, Characterizing data complexity for conjunctive query answering in expressive description logics, in: Proceedings of the 21st National Conference on Artificial Intelligence - Volume 1, AAAI'06, AAAI Press, 2006, p. 275–280.

[5] M. Ortiz, D. Calvanese, T. Eiter, Data complexity of query answering in expressive description logics via tableaux, J. Autom. Reason. 41 (2008) 61–98. doi:10.1007/s10817-008-9102-9.

[6] M. Schmidt-Schauß, G. Smolka, Attributive concept descriptions with complements, Artificial Intelligence 48 (1991) 1–26. doi:10.1016/0004-3702(91)90078-X.

[7] Y. Kazakov, B. Motik, A resolution-based decision procedure for $\mathcal{SHOIQ}$, in: U. Furbach, N. Shankar (Eds.), Automated Reasoning, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 662–677.

[8] B. Motik, U. Sattler, A comparison of reasoning techniques for querying large description logic aboxes, in: M. Hermann, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 227–241.

[9] D. Trivela, G. Stoilos, A. Chortaras, G. Stamou, Optimising resolution-based rewriting algorithms for owl ontologies, Journal of Web Semantics 33 (2015) 30–49. doi:10.1016/j.websem.2015.02.001, ontology-based Data Access.

[10] F. Simancik, Y. Kazakov, I. Horrocks, Consequence-based reasoning beyond horn ontologies, in: IJCAI, 2011, pp. 1093–1099.

[11] Y. Kazakov, Consequence-driven reasoning for horn shiq ontologies, in: Proceedings of the 21st International Jont Conference on Artifical Intelligence, IJCAI'09, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2009, p. 2040–2045.

[12] S. Negri, J. Von Plato, Proof analysis: a contribution to HIlbert's last problem, Cambridge University Press, 2011.

[13] A. Borgida, E. Franconi, I. Horrocks, Explaining alc subsumption, in: Proceedings of the 14th European Conference on Artificial Intelligence, ECAI'00, IOS Press, NLD, 2000, p. 209–213.

[14] M. Hofmann, Proof-theoretic approach to description-logic, in: 20th Annual IEEE Symposium on Logic in Computer Science (LICS' 05), 2005, pp. 229–237. doi:10.1109/LICS.2005.38.

[15] A. Rademaker, A Proof Theory for Description Logics, Springer Science & Business Media, 2012.

[16] U. Straccia, A sequent calculus for reasoning in four-valued description logics, in: D. Galmiche (Ed.), Automated Reasoning with Analytic Tableaux and Related Methods, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997, pp. 343–357.

[17] B. C. Grau, I. Horrocks, B. Motik, B. Parsia, P. Patel-Schneider, U. Sattler, Owl 2: The next step for owl, Journal of Web Semantics 6 (2008) 309–322. doi:10.1016/j.websem.2008.05.001, semantic Web Challenge 2006/2007.

[18] M. Ortiz, M. Šimkus, Reasoning and query answering in description logics, in: T. Eiter, T. Krennwallner (Eds.), Reasoning Web. Semantic Technologies for Advanced Query Answering: 8th International Summer School 2012, Vienna, Austria, September 3-8, 2012. Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 1–53. doi:10.1007/978-3-642-33158-9_1.

[19] S. C. Kleene, Introduction to Metamathematics, American Elsevier Publishing Company, INC. - New York, 1952.

[20] A. K. Simpson, The Proof Theory and Semantics of Intuitionistic Modal Logic, Ph.D. thesis, University of Edinburgh. College of Science and Engineering. School of Informatics, 1994.

[21] L. Viganò, Labelled Non-Classical Logics, Springer Science & Business Media, 2000.

[22] S. A. Kripke, A completeness theorem in modal logic, Journal of Symbolic Logic 24 (1959) 1–14. doi:10.2307/2964568.

[23] T. Lyon, K. van Berkel, Automating agential reasoning: Proof-calculi and syntactic decidability for stit logics, in: M. Baldoni, M. Dastani, B. Liao, Y. Sakurai, R. Zalila Wenkstern (Eds.), PRIMA 2019: Principles and Practice of Multi-Agent Systems - 22nd International Conference, Proceedings, volume 11873 of *Lecture Notes in Computer Science*, Springer International Publishing, Cham, 2019, pp. 202–218.

[24] T. Lyon, Refining Labelled Systems for Modal and Constructive Logics with Applications, Ph.D. thesis, Technische Universität Wien, 2021.

[25] T. Lyon, A. Tiu, R. Goré, R. Clouston, Syntactic interpolation for tense logics and bi-intuitionistic logic via nested sequents, in: M. Fernández, A. Muscholl (Eds.), 28th EACSL Annual Conference on Computer Science Logic, CSL 2020, volume 152 of *LIPIcs*, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020, pp. 28:1–28:16. doi:10.4230/LIPIcs.CSL.2020.28.

[26] M. Ferrari, C. Fiorentini, G. Fiorino, $\mathcal{BCDL}$: Basic constructive description logic, J. Autom. Reason. 44 (2010) 371–399. doi:10.1007/s10817-009-9160-7.

[27] V. de Paiva, Constructive description logics: What , why and how, 2006. Unpublished.

[28] S. Scheele, Model and Proof Theory of Constructive ALC, Constructive Description Logics, University of Bamberg Press, 2015.

[29] B. Glimm, I. Horrocks, B. Motik, G. Stoilos, Z. Wang, Hermit: an owl 2 reasoner, Journal of Automated Reasoning 53 (2014) 245–269.

[30] S. Maehara, On the interpolation theorem of craig, Sûgaku 12 (1960) 235–237.