Constructive Interpolation and Concept-Based Beth Definability for Description Logics via Sequents

Tim S. Lyon, **Jonas Karge**
Institute of Artificial Intelligence, Technische Universität Dresden
{timothy_stephen.lyon, jonas.karge}@tu-dresden.de

Abstract

We introduce a constructive method applicable to a large number of description logics (DLs) for establishing the concept-based Beth definability property (CBP) based on sequent systems. Using the highly expressive DL $\mathcal{RIQ}$ as a case study, we introduce novel sequent calculi for $\mathcal{RIQ}$-ontologies and show how certain interpolants can be computed from sequent calculus proofs, which permit the extraction of explicit definitions of implicitly definable concepts. To the best of our knowledge, this is the first sequent-based approach to computing interpolants and definitions within the context of DLs, as well as the first proof that $\mathcal{RIQ}$ enjoys the CBP. Moreover, due to the modularity of our sequent systems, our results hold for any restriction of $\mathcal{RIQ}$, and are applicable to other DLs by suitable modifications.

1 Introduction

Defining new concepts in terms of given concepts and relations is an important operation within the context of description logics (DLs), and logic more generally. Typically, a new concept NewC can be defined in one of two ways: (1) *implicitly*, by specifying a set of axioms such that the interpretation of NewC is uniquely determined by the interpretation of the given concepts and relations, or (2) *explicitly*, by writing a definition $\mathsf{NewC} \equiv \mathsf{D}$ where NewC does not appear in D. Description logics for which implicit definability implies explicit definability are said to be *definitorially complete* [Baader and Nutt, 2003; ten Cate *et al.*, 2006], or to exhibit the *concept-based Beth definability property (CBP)* [ten Cate *et al.*, 2013]. This is Beth's definability property [Beth, 1956] from first-order logic rephrased for DLs.

Beth definability and variations thereof have found numerous applications in DLs. For example, the property has been used in ontology engineering to extract acyclic terminologies from general ones [Baader and Nutt, 2003; ten Cate *et al.*, 2006]. This is of particular importance since reasoning with acyclic terminologies is usually less complex than with general ones, e.g. satisfiability over acyclic $\mathcal{ALC}$-terminologies is PSPACE-complete while being EXPTIME-complete over general $\mathcal{ALC}$-terminologies [Donini, 2003]. Other applications include, rewriting ontology-mediated queries [Franconi and Kerhet, 2019; Seylan *et al.*, 2009; Toman and Weddell, 2022], learning concepts separating positive and negative examples [Artale *et al.*, 2023; Funk *et al.*, 2019], and computing referring expressions, which is of value in computational linguistics and data management [Areces *et al.*, 2008; Borgida *et al.*, 2016; Artale *et al.*, 2021].

A number of methods have been used to confirm the existence of, or actually compute, explicit definitions of implicitly definable concepts for expressive DLs; e.g. model-theoretic mosaic-based methods have been employed to decide the existence of explicit definitions for $\mathcal{ALCH}$, $\mathcal{ALCO}$, and $\mathcal{ALCHOI}$ [Artale *et al.*, 2023; Jung *et al.*, 2022]. However, as noted in these works, these methods are *non-constructive*, confirming the existence of explicit definitions without necessarily providing them. Thus, interest has been expressed in developing constructive methods that *actually compute* explicit definitions. We note that constructive methods have been employed in the literature, e.g. methods relying on the computation of normal forms and uniform interpolants [ten Cate *et al.*, 2006] or which compute explicit definitions using tableau-based algorithms [ten Cate *et al.*, 2013]. With the aim of furthering this programme, we present a constructive method applicable to a large number of DLs, which computes explicit concept-based definitions of implicitly definable concepts and establishes the CBP by means of *sequent systems*.

Since its introduction in the 1930's, Gentzen's sequent calculus has become one of the preferred formalisms for the construction of proof calculi [Gentzen, 1935a; Gentzen, 1935b]. A sequent calculus is a set of inference rules operating over expressions (called *sequents*) of the form $\Gamma \vdash \Delta$ with Γ and Δ sequences or (multi)sets of formulae. Sequent systems have found fruitful applications, being exploited in the development of automated reasoning methods [Slaney, 1997] and being used to establish non-trivial properties of logics such as consistency [Gentzen, 1935a; Gentzen, 1935b], decidability [Dyckhoff, 1992], and interpolation [Maehara, 1960]. Regarding this last point, it was first shown by Maehara that sequent systems could be leveraged to *constructively* prove the Craig interpolation property [Craig, 1957] of a logic. Since this seminal work, Maehara's interpolation method has been extended and adapted in a variety of ways to prove Craig interpolation for diverse classes of logics with sequent-style systems, including modal logics [Fitting and Kuznets, 2015], intermediate logics [Kuznets and Lellmann, 2018], and temporal logics [Lyon *et al.*, 2020]. As Craig interpolation implies Beth definability, it follows that the sequent-based methodology is applicable to the latter.

In this paper, we provide the first sequent calculi for $\mathcal{RIQ}$-ontologies and show how these calculi can be used to compute interpolants, explicit definitions, and to confirm the CBP. Although our work is inspired by Maehara's method, we note that it is a non-trivial generalization of that method. As discussed in [Lyon *et al.*, 2020], Maehara's original method is quite restricted, being inapplicable in many cases to even basic modal logics, which *a fortiori* means the method is inapplicable to expressive DLs. To overcome these difficulties, we use a generalized notion of sequent and interpolant that encodes a tree whose nodes are multisets of DL concepts accompanied by (in)equalities over nodes. Given a proof with such sequents, we show that all axiomatic sequents can be assigned interpolants—which are themselves sequents—and that such interpolants can be 'propagated' through the proof yielding an interpolant of the conclusion. Explicit definitions can then be readily extracted from these interpolants. We note that our method is constructive in the sense that interpolants are computed relative to a *given proof* of a general concept inclusion implied by a $\mathcal{RIQ}$-ontology. Although such proofs are in principle computable, we left the specification of an explicit proof-search algorithm that builds such proofs to future work, noting that such algorithms can be written by adapting known techniques; e.g. [Horrocks and Sattler, 2004].

Finally, we remark that although our work shares similarities with that of [ten Cate *et al.*, 2013], our method goes beyond this work as we establish the CBP for the highly expressive DL $\mathcal{RIQ}$, and due to the modularity of our sequent systems, our method is applicable to any restriction of $\mathcal{RIQ}$, e.g. logics within the $\mathcal{FL}$ and $\mathcal{EL}$ families [Donini *et al.*, 1997; Baader *et al.*, 2005]. By *modularity* we mean that the deletion of inference rules or modification of side conditions on rules allows for sequent systems to be provided for fragments of $\mathcal{RIQ}$. Our work also intersects that of [ten Cate *et al.*, 2006], which establishes the CBP for $\mathcal{ALC}$ extended with PUR Horn conditions, but differs both in terms of methodology and our consideration of qualified number restrictions.

Outline of Paper. In Section 2, we define the logic $\mathcal{RIQ}$, define the CBP and related notions, and explicate certain grammar theoretic concepts used in formulating inference rules. In Section 3, we present our sequent systems and establish that each system enjoys fundamental properties. Section 4 develops and explains our new sequent-based method that computes explicit definitions of implicitly definable concepts and establishes the CBP, using $\mathcal{RIQ}$ as case study. To the best of our knowledge, this is the first proof that $\mathcal{RIQ}$ has the CBP. Section 5 concludes and discusses future work. Last, we remark that proofs of all results can be found in the online, appended version [Lyon and Karge, 2024].

2 Preliminaries

In the first part of this section, we introduce the language and semantics for the description logic $\mathcal{RIQ}$, which subsumes various expressive and lightweight DLs [Baader *et al.*, 2005; Calvanese and De Giacomo, 2003]. Subsequently, we discuss and define a notion of interpolation and concept-based Beth definability, which will be of pivotal interest in this paper. In the last part of this section, we introduce special types of semi-Thue systems [Post, 1947], referred to as $\mathbf{R}$-*systems*, which are essential in the formulation of our sequent calculi.

2.1 Language and Semantics: $\mathcal{RIQ}$

The description logic $\mathcal{RIQ}$ is defined relative to a *vocabulary* $\mathcal{V} = (\mathsf{N_R}, \mathsf{N_C})$, which is a pair containing pairwise disjoint, countable sets. The set $\mathsf{N_R}$ contains *role names* used to denote binary relations and the set $\mathsf{N_C}$ contains *concept names* used to denote classes of entities. We use the (potentially annotated) symbols $r, s, \ldots$ to denote role names, and $A, B, \ldots$ to denote concept names. We define a *role* to be a role name or an *inverse role* r^- such that $r \in \mathsf{N_R}$. We define the *inverse* of a role to be $\mathsf{Inv}(r) = r^-$ and $\mathsf{Inv}(r^-) = r$ given that $r \in \mathsf{N_R}$. We let $\mathbf{R} := \mathsf{N_R} \cup \{\mathsf{Inv}(r) \mid r \in \mathsf{N_R}\}$ denote the set of roles.

A *complex role inclusion axiom (RIA)* is an expression $r_1 \circ \cdots \circ r_n \sqsubseteq s$ such that $r_1, \ldots, r_n$ and s are roles, and $\circ$ denotes the usual composition operation over binary relations; we assume n-ary compositions $r_1 \circ \cdots \circ r_n$ associate to the left. We define an *RBox* $\mathcal{R}$ to be a finite collection of RIAs. As identified by Horrocks and Sattler [Horrocks and Sattler, 2004], to ensure the decidability of reasoning with $\mathcal{RIQ}$, only *regular RBoxes* may be used in ontologies (defined below).[1] Let $\prec$ be a strict partial order on the set $\mathsf{N_R}$ of role names; we define an RIA $w \sqsubseteq r$ to be $\prec$-*regular iff* r is a role name, and either (1) $w = rr$, (2) $w = r^-$, (3) $w = s_1 \circ \cdots \circ s_n$ and $s_i \prec r$ for all $1 \leq i \leq n$, (4) $w = r \circ s_1 \circ \cdots \circ s_n$ and $s_i \prec r$ for all $1 \leq i \leq n$, or (5) $w = s_1 \circ \cdots \circ s_n \circ r$ and $s_i \prec r$ for all $1 \leq i \leq n$. An RBox $\mathcal{R}$ is defined to be *regular iff* a strict partial order $\prec$ over $\mathsf{N_R}$ exists such that every RIA in $\mathcal{R}$ is $\prec$-regular.

We recursively define a role name r to be *simple* (with respect to an RBox $\mathcal{R}$) *iff* either (1) no RIA of the form $w \sqsubseteq r$ occurs in $\mathcal{R}$, or (2) for each $s \sqsubseteq r \in \mathcal{R}$, s is a simple role name or its inverse is. Also, an inverse role r^- is defined to be *simple* if r is simple.

We define *complex concepts* to be formulae in negation normal form generated by the following grammar in BNF:

$$C ::= A \mid \neg A \mid (C \odot C) \mid (Q.C) \mid (\leqslant ns.C) \mid (\geqslant ns.C)$$

where $A \in \mathsf{N_C}$, $\odot \in \{\sqcup, \sqcap\}$, $Q \in \{\exists r, \forall r \mid r \in \mathbf{R}\}$, s is a simple role, and $n \in \mathbb{N}$. We use the symbols C, D, ... (potentially annotated) to denote complex concepts. We define $\top = A \sqcup \neg A$ and $\bot = A \sqcap \neg A$ for a fixed $A \in \mathsf{N_C}$, and define a *literal* L to be either a concept name or its negation, i.e. $L \in \{A, \neg A \mid A \in \mathsf{N_C}\}$. For a concept name A, we define $\dot{\neg} A := \neg A$ and $\dot{\neg}\neg A := A$, and we lift the definition of negation to complex concepts in the usual way, noting that $\dot{\neg}(\leqslant nr.C) := (\geqslant (n{+}1) r.C)$, and

$$\dot{\neg}(\geqslant nr.C) := \begin{cases} \bot & \text{if } n = 0, \\ \leqslant (n{-}1) r.C & \text{otherwise.} \end{cases}$$

[1] Note, our interpolation results go through for general RBoxes, i.e. this restriction is not needed for the work in Sections 3 and 4.

We recursively define the *weight* of a concept C as follows: (1) $w(L) = 1$ with $L \in \{A, \neg A \mid A \in \mathsf{N_C}\}$, (2) $w(C \odot D) = w(C)+w(D)+1$ with $\odot \in \{\sqcap, \sqcup\}$, (3) $w(Q.C) = w(C)+1$ with $Q \in \{\exists r, \forall r \mid r \in \mathbf{R}\}$, (4) $w(\leqslant ns.C) = w(C) + n + 1$, and (5) $w(\geqslant ns.C) = w(C) + n$.

A *general concept inclusion axiom (GCI)* is a formula of the form $C \sqsubseteq D$ such that C and D are complex concepts. A *TBox* $\mathcal{T}$ is a finite set of GCIs and we make the simplifying assumption that every GCI in a TBox $\mathcal{T}$ is of the form $\top \sqsubseteq C$. We define a $\mathcal{RIQ}$*-ontology* $\mathcal{O}$ (which we refer to as an *ontology* for short) to be the union of an RBox $\mathcal{R}$ and TBox $\mathcal{T}$, that is, $\mathcal{O} = \mathcal{R} \cup \mathcal{T}$. For a set X of concepts, GCIs, or RIAs, we let $\mathrm{con}(X)$ denote the set of all concept names occurring in X, and we let $\mathrm{sig}(X)$ denote the set of all concept names and roles occurring in X. Symbols from a vocabulary $\mathcal{V}$ are interpreted accordingly:

Definition 1 (Interpretation)**.** *An* interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *is a pair consisting of a non-empty set* $\Delta^{\mathcal{I}}$ *called the* domain *and a map* $\cdot^{\mathcal{I}}$ *such that*

- *if* $A \in \mathsf{N_C}$*, then* $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ *with* $\neg A^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus A^{\mathcal{I}}$*;*
- *if* $r \in \mathsf{N_R}$*, then* $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$*.*

We define $(r^-)^{\mathcal{I}} = \{(b, a) \mid (a, b) \in r^{\mathcal{I}}\}$ *and interpret compositions over roles in the usual way. We lift interpretations to complex concepts accordingly:*

- $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$*;*
- $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$*;*
- $\exists r.C^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \exists b \in \Delta^{\mathcal{I}}, (a, b) \in r^{\mathcal{I}} \ \& \ b \in C^{\mathcal{I}}\}$*;*
- $\forall r.C^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \forall b \in \Delta^{\mathcal{I}}, (a, b) \in r^{\mathcal{I}} \Rightarrow b \in C^{\mathcal{I}}\}$*;*
- $\leqslant ns.C^{\mathcal{I}} := \{a \in \Delta^{\mathcal{I}} : |\{b : (a, b) \in s^{\mathcal{I}} \ \& \ b \in C^{\mathcal{I}}\}| \leq n\}$*;*
- $\geqslant ns.C^{\mathcal{I}} := \{a \in \Delta^{\mathcal{I}} : |\{b : (a, b) \in s^{\mathcal{I}} \ \& \ b \in C^{\mathcal{I}}\}| \geq n\}$*.*

An interpretation satisfies $C \sqsubseteq D$ *or* $r_1 \circ \cdots \circ r_n \sqsubseteq s$*, written* $\mathcal{I} \vDash C \sqsubseteq D$ *and* $\mathcal{I} \vDash r_1 \circ \cdots \circ r_n \sqsubseteq s$ *iff* $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ *and* $(r_1 \circ \cdots \circ r_n)^{\mathcal{I}} \subseteq s^{\mathcal{I}}$*, respectively. An interpretation* $\mathcal{I}$ *is defined to be a* model *of an ontology* $\mathcal{O}$*, written* $\mathcal{I} \vDash \mathcal{O}$*, iff it satisfies all GCIs and RIAs in* $\mathcal{O}$*. We write* $\mathcal{O} \vDash C \sqsubseteq D$ *iff for every interpretation* $\mathcal{I}$*, if* $\mathcal{I} \vDash \mathcal{O}$*, then* $\mathcal{I} \vDash C \sqsubseteq D$*, and we write* $\mathcal{O} \vDash C \equiv D$ *when* $\mathcal{O} \vDash C \sqsubseteq D$ *and* $\mathcal{O} \vDash D \sqsubseteq C$*.*

2.2 Definability and Interpolation

The notion of Beth definability, first defined within the context of first-order logic [Beth, 1956], takes on a number of distinct formulations within the context of DLs. In [Baader and Nutt, 2003; ten Cate *et al.*, 2006], Beth definability is reinterpreted as the notion of *definitorial completeness*, which has also been named *concept-based Beth definability (CBP)* [ten Cate *et al.*, 2013]. Intuitively, a DL $\mathcal{L}$ has the CBP when the implicit definability of a concept C under an $\mathcal{L}$-ontology $\mathcal{O}$ using a signature $\Theta = \Xi \cup \mathsf{N_R}$ with $\Xi \subseteq \mathrm{con}(C, \mathcal{O})$ implies its explicit definability using symbols from Θ. This is distinct from the *projective Beth definability property (PBDP)*, which is defined in the same way but relative to a signature $\Theta \subseteq \mathrm{sig}(C, \mathcal{O})$, or the weaker *Beth definability property (BDP)* where the signature Θ is the set of all symbols distinct from the concept defined [Artale *et al.*, 2023]. In this paper, we focus on the CBP, and leave the investigation of sequent-based methodologies for establishing other definability properties to future work. Let us now formally define the CBP.

Let $\mathcal{L}$ be a DL, C be a complex concept in $\mathcal{L}$, $\mathcal{O}$ an $\mathcal{L}$-ontology, and $\Theta \subseteq \mathrm{con}(C, \mathcal{O})$.[2] We define C to be *implicitly concept-definable* from Θ under $\mathcal{O}$ *iff* for any two models $\mathcal{I}$ and $\mathcal{J}$ of $\mathcal{O}$ such that $\Delta^{\mathcal{I}} = \Delta^{\mathcal{J}}$ and for each $P \in \Theta \cup \mathsf{N_R}$, $P^{\mathcal{I}} = P^{\mathcal{J}}$, it follows that $C^{\mathcal{I}} = C^{\mathcal{J}}$. We remark that this notion can be reformulated as a standard reasoning problem, that is, C is implicitly concept-definable from Θ under $\mathcal{O}$ *iff*

$$\mathcal{O} \cup \mathcal{O}_{\Theta} \vDash C \sqsubseteq C_{\Theta} \tag{1}$$

where $\mathcal{O}_{\Theta}$ and C_{Θ} are obtained from $\mathcal{O}$ and C, respectively, by uniformly replacing every concept name $A \notin \Theta$ by a fresh concept name. We define C to be *explicitly concept-definable* from Θ under $\mathcal{O}$ *iff* there exists a complex concept D (called an *explicit concept-definition*) such that $\mathcal{O} \vDash C \equiv D$ and $\mathrm{con}(D) \subseteq \Theta$.

Definition 2 (Concept-Based Beth Definability)**.** *Let* $\mathcal{L}$ *be a DL,* C *be a complex concept in* $\mathcal{L}$*,* $\mathcal{O}$ *be an* $\mathcal{L}$*-ontology, and* $\Theta \subseteq \mathrm{con}(C, \mathcal{O})$*. We say that* $\mathcal{L}$ *has the* concept-name Beth definability property (CBP) *iff if* C *is implicitly concept-definable from* Θ *under* $\mathcal{O}$*, then* C *is explicitly concept-definable from* Θ *under* $\mathcal{O}$*.*

It is typical to establish definability properties by means of an interpolation theroem (cf. [ten Cate *et al.*, 2013; Craig, 1957; Jung *et al.*, 2022]). We therefore define a suitable notion of interpolation that implies the CBP, which we call *concept interpolation*.

Definition 3 (Concept Interpolation Property)**.** *Let* $\mathcal{L}$ *be a DL,* $\mathcal{O}_1$ *and* $\mathcal{O}_2$ *be* $\mathcal{L}$*-ontologies with* $\mathcal{O} = \mathcal{O}_1 \cup \mathcal{O}_2$*, and* C *and* D *be* $\mathcal{L}$*-concepts. We define an* $\mathcal{L}$*-concept* I *to be a* concept interpolant *for* $C \sqsubseteq D$ *under* $\mathcal{O}$ *iff (1)* $\mathrm{con}(I) \subseteq \mathrm{con}(\mathcal{O}_1, C) \cap \mathrm{con}(\mathcal{O}_2, D)$*, (2)* $\mathcal{O} \vDash C \sqsubseteq I$*, and (3)* $\mathcal{O} \vDash I \sqsubseteq D$*. A DL* $\mathcal{L}$ *enjoys the* concept interpolation property *if for all* $\mathcal{L}$*-ontologies* $\mathcal{O}_1$*,* $\mathcal{O}_2$ *with* $\mathcal{O} = \mathcal{O}_1 \cup \mathcal{O}_2$ *and* $\mathcal{L}$*-concepts* C*,* D *such that* $\mathcal{O} \vDash C \sqsubseteq D$ *there exists a concept interpolant for* $C \sqsubseteq D$ *under* $\mathcal{O}$*.*

Lemma 1. *If a DL* $\mathcal{L}$ *enjoys the concept interpolation property, then it enjoys the CBP.*

2.3 R-Systems

We let $\mathbf{R}$ serve as our *alphabet* with each role serving as a *character*. The set $\mathbf{R}^*$ of *strings over* $\mathbf{R}$ is defined to be the smallest set satisfying the following conditions: (i) $\mathbf{R} \cup \{\varepsilon\} \subseteq \mathbf{R}^*$ with ε the *empty string*, and (ii) If $S \in \mathbf{R}^*$ and $r \in \mathbf{R}$, then $Sr \in \mathbf{R}^*$, where Sr represents the *concatenation* of S and r. We use $S, R, \ldots$ (potentially annotated) to denote strings from $\mathbf{R}^*$, and we have $S\varepsilon = \varepsilon S = S$, for the empty string ε. The inverse operation on strings is defined as: (1) $\mathsf{Inv}(\varepsilon) := \varepsilon$, and (2) If $S = r_1 \cdots r_n$, then $\mathsf{Inv}(S) := \mathsf{Inv}(r_n) \cdots \mathsf{Inv}(r_1)$.

We now define $\mathbf{R}$*-systems*, which are special types of *Semi-Thue systems* [Post, 1947], relative to ontologies. These will permit us to derive strings of roles from a given role and encode the information present in a given ontology.

[2] In this paper, we take a DL $\mathcal{L}$ to be $\mathcal{RIQ}$ or a fragment thereof.

Definition 4 ($\mathbf{R}$-system)**.** *Let $\mathcal{O}$ be an ontology. We define the* $\mathbf{R}$-system $\mathrm{G}(\mathcal{O})$ *to be the smallest set of* production rules *of the form $r \longrightarrow S$, where $r \in \mathbf{R}$ and $S \in \mathbf{R}^*$, such that if $r_1 \circ \cdots \circ r_n \sqsubseteq s \in \mathcal{O}$, then*

$$(s \longrightarrow r_1 \cdots r_n), (\mathsf{Inv}(s) \longrightarrow \mathsf{Inv}(r_n) \cdots \mathsf{Inv}(r_1)) \in \mathrm{G}(\mathcal{O}).$$

Definition 5 (Derivation, Language)**.** *Let $\mathcal{O}$ be an ontology and $\mathrm{G}(\mathcal{O})$ be its $\mathbf{R}$-system. We write $S \longrightarrow_{\mathrm{G}(\mathcal{O})} R$ and say that the string R may be derived from the string S in* one-step *iff there are strings $S', R' \in \mathbf{R}^*$ and $r \longrightarrow T \in \mathrm{G}(\mathcal{O})$ such that $S = S'rR'$ and $R = S'TR'$. We define the* derivation relation $\longrightarrow^*_{\mathrm{G}(\mathcal{O})}$ *to be the reflexive and transitive closure of $\longrightarrow_{\mathrm{G}(\mathcal{O})}$. For $S, R \in \mathbf{R}^*$, we call $S \longrightarrow^*_{\mathrm{G}(\mathcal{O})} R$ a* derivation of R from S, *and define the* length *of a derivation to be the minimal number of one-step derivations required to derive R from S in $\mathrm{G}(\mathcal{O})$. Last, we define the* language $L_{\mathrm{G}(\mathcal{O})}(r) := \{S \mid r \longrightarrow^*_{\mathrm{G}(\mathcal{O})} S\}$*, where $r \in \mathbf{R}$.*

3 Sequent Systems

We let $\mathrm{Lab} = \{x, y, z, \ldots\}$ be a countably infinite set of labels, define a *role atom* to be an expression of the form $r(x, y)$ with $r \in \mathbf{R}$ and $x, y \in \mathrm{Lab}$, define an *equality atom* and *inequality atom* to be an expression of the form $x \doteq y$ and $x \not\doteq y$ with $x, y \in \mathrm{Lab}$, respectively, and define a *labeled concept* to be an expression $x : C$ with $x \in \mathrm{Lab}$ and C a complex concept. We refer to role, equality, and inequality atoms as *structural atoms* more generally. For a (multi)set X and Y of structural atoms and/or labeled concepts, we let X, Y represent their union and let $\mathrm{Lab}(X)$ be the set of labels occurring therein. We say that a set Γ of structural atoms *forms a tree iff* the graph $T(\Gamma) = (V, E)$ is a directed tree with $V = \mathrm{Lab}(\Gamma)$, and $(x, y) \in E$ *iff* $r(x, y) \in \Gamma$. A *sequent* is defined to be an expression of the form $S := \Gamma \vdash \Delta$ such that (1) Γ is a set of structural atoms that forms a tree, (2) Δ is a multiset of labeled concepts, (3) if $\Gamma \neq \emptyset$, then $\mathrm{Lab}(\Delta) \subseteq \mathrm{Lab}(\Gamma)$, and (4) if $\Gamma = \emptyset$, then $|\mathrm{Lab}(\Delta)| = 1$. In a sequent $\Gamma \vdash \Delta$, we refer to Γ as the *antecedent*, Δ as the *consequent*, and we define $\Delta{\restriction}x := \{C \mid x : C \in \Delta\}$.

Recall that every GCI in an ontology $\mathcal{O}$ is assumed to be of the form $\top \sqsubseteq C$. For an ontology $\mathcal{O} = \mathcal{R} \cup \mathcal{T}$ and label $x \in \mathrm{Lab}$, we let $x : \dot{\neg}\mathcal{T}_{\mathcal{O}} = x : \dot{\neg}C_1, \ldots, x : \dot{\neg}C_n$ such that $\mathcal{T} = \{\top \sqsubseteq C_1, \ldots, \top \sqsubseteq C_n\}$. For labels $x_1, \ldots, x_n \in \mathrm{Lab}$, we define $\Gamma^{\not\doteq}(x_1, \ldots, x_n) = \{x_i \not\doteq x_j \mid 1 \leq i < j \leq n\}$. We let $x \approx y \in \{x \doteq y, y \doteq x\}$ and write $x =^*_\Gamma y$ *iff* there exist $z_1, \ldots, z_n \in \mathrm{Lab}(\Gamma)$ such that $z_1 \approx z_2, \ldots, z_{n-1} \approx z_n$ with $x = z_1$ and $y = z_n$. We make use of equivalence classes of labels in the formulation of certain inference rules below and define $[x]_\Gamma := \{y \mid x =^*_\Gamma y\}$ for a sequent $\Gamma \vdash \Delta$.

A uniform presentation of our sequent systems in presented in Figure 1. We note that each sequent calculus $\mathsf{S}(\mathcal{O})$ takes a $\mathcal{RIQ}$-ontology $\mathcal{O}$ as an input parameter, which determines the functionality of certain inference rules depending on the contents of $\mathcal{O}$. The calculus $\mathsf{S}(\mathcal{O})$ contains the *initial rules* (id) and $(id_{\doteq})$, which generate axioms that are used to begin a proof, the *logical rules* $(\sqcup)$, $(\sqcap)$, $(\exists r)$, $(\forall r)$, $(\leqslant nr)$, and $(\geqslant nr)$, which introduce complex concepts, and the *substitution rule* $(s_{\doteq})$. We note that $A \in \mathsf{N_C}$ in the (id) rule and L is a literal in the $(s_{\doteq})$ rule. The $(id_{\doteq})$ and $(s_{\doteq})$ rules are subject to a side condition, namely, each rule is applicable only if $x =^*_\Gamma y$. The $(\forall r)$ and $(\leqslant nr)$ rules are subject to side conditions as well: the label y and the labels $y_0, \ldots, y_n$ must be *fresh* in $(\forall r)$ and $(\leqslant nr)$, respectively, meaning such labels may not occur in the conclusion of a rule application. Last, we note that the $(\exists r)$ and $(\geqslant nr)$ rules are special types of logical rules, referred to as *propagation rules*; cf. [Castilho *et al.*, 1997; Fitting, 1972]. These rules operate by viewing sequents as types of automata, referred to as *propagation graphs*, which bottom-up propagate formulae along special paths, referred to as *propagation paths* (see Example 1 below).

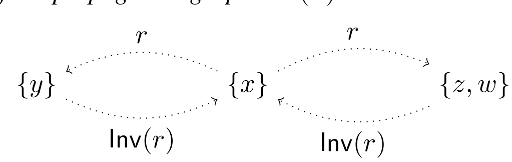
Definition 6 (Propagation Graph)**.** *We define the* propagation graph $\mathrm{PG}(\Gamma) = (V, E)$ *of a sequent $\Gamma \vdash \Delta$ such that $[x]_\Gamma \in V$ iff $x \in \mathrm{Lab}(\Gamma)$, and $([x]_\Gamma, r, [y]_\Gamma), ([y]_\Gamma, \mathsf{Inv}(r), [x]_\Gamma) \in E$ iff there exist $z \in [x]_\Gamma$ and $w \in [y]_\Gamma$ such that $r(z, w) \in \Gamma$. If we write $[x]_\Gamma \in \mathrm{PG}(\Gamma)$, then we mean $[x]_\Gamma \in V$, and if we write $([x]_\Gamma, r, [y]_\Gamma) \in \mathrm{PG}(\Gamma)$, we mean $([x]_\Gamma, r, [y]_\Gamma) \in E$.*

We note that our propagation graphs are generalizations of those employed in sequent systems for modal and non-classical logics [Ciabattoni *et al.*, 2021; Goré *et al.*, 2011; Lyon, 2021]. In particular, due to the inclusion of equality atoms, we must define propagation graphs over equivalence classes of labels, rather than over labels themselves. This lets us define novel and correct propagation rules in the presence of (in)equalities and counting quantifiers.

Definition 7 (Propagation Path)**.** *Given a propagation graph $\mathrm{PG}(\Gamma) = (V, E)$, $[x]_\Gamma, [y]_\Gamma \in V$, and $r \in \mathbf{R}$, we write $\mathrm{PG}(\Gamma) \models [x]_\Gamma \overset{r}{\rightsquigarrow} [y]_\Gamma$ iff $([x]_\Gamma, r, [y]_\Gamma) \in E$. Given a string $rS \in \mathbf{R}^*$ where $r \in \mathbf{R}$, we define $\mathrm{PG}(\Gamma) \models [x]_\Gamma \overset{rS}{\rightsquigarrow} [y]_\Gamma$ as '$\exists_{[z]_\Gamma \in V}\ \mathrm{PG}(\Gamma) \models [x]_\Gamma \overset{r}{\rightsquigarrow} [z]_\Gamma$ and $\mathrm{PG}(\Gamma) \models [z]_\Gamma \overset{S}{\rightsquigarrow} [y]_\Gamma$', and we take $\mathrm{PG}(\Gamma) \models [x]_\Gamma \overset{\varepsilon}{\rightsquigarrow} [y]_\Gamma$ to mean that $[x]_\Gamma = [y]_\Gamma$. Additionally, when $\mathrm{PG}(\Gamma)$ is clear from the context we may simply write $[x]_\Gamma \overset{S}{\rightsquigarrow} [y]_\Gamma$ to express $\mathrm{PG}(\Gamma) \models [x]_\Gamma \overset{S}{\rightsquigarrow} [y]_\Gamma$. Finally, given a language $L_{\mathrm{G}(\mathcal{O})}(r)$ of some $\mathbf{R}$-system $\mathrm{G}(\mathcal{O})$ and $r \in \mathbf{R}$, we use $[x]_\Gamma \overset{L}{\rightsquigarrow} [y]_\Gamma$ with $L = L_{\mathrm{G}(\mathcal{O})}(r)$ iff there is a string $S \in L_{\mathrm{G}(\mathcal{O})}(r)$ such that $[x]_\Gamma \overset{S}{\rightsquigarrow} [y]_\Gamma$.*

To provide intuition concerning the functionality of propagation rules, we illustrate a (bottom-up) application of $(\geqslant nr)$.

Example 1. *Let us consider the sequent $\Gamma \vdash x : \geqslant 2r.C$ with $\Gamma = r(x, y), r(x, z), r(x, w), z \doteq w$. A pictorial representation of the propagation graph $\mathrm{PG}(\Gamma)$ is shown below.*



One can see that there are two labels y and z such that $[x]_\Gamma \overset{r}{\rightsquigarrow} [y]_\Gamma$ and $[x]_\Gamma \overset{r}{\rightsquigarrow} [z]_\Gamma$. Note that $r \in L_{\mathrm{G}(\mathcal{O})}(r)$ by definition. Therefore, we may (bottom-up) apply the $(\geqslant nr)$ rule to obtain the three premises $\Gamma \vdash x : \geqslant 2r.C, y : C$, $\Gamma \vdash x : \geqslant 2r.C, z : C$, and $\Gamma, y \doteq z \vdash x : \geqslant 2r.C$.

We define a *proof* in $\mathsf{S}(\mathcal{O})$ inductively: (1) each instance of an initial rule (r), as shown below left, is a proof with conclusion S, and (2) if n proofs exist with the respective

$$\frac{}{\Gamma \vdash x : A, x : \neg A, \Delta}\,(id) \qquad \frac{}{\Gamma, x \not\doteq y \vdash \Delta}\,(id_{\doteq})^{\dagger_1} \qquad \frac{\Gamma \vdash x : L, y : L, \Delta}{\Gamma \vdash x : L, \Delta}\,(s_{\doteq})^{\dagger_1} \qquad \frac{\Gamma \vdash x : C, x : D, \Delta}{\Gamma \vdash x : C \sqcup D, \Delta}\,(\sqcup)$$

$$\frac{\Gamma \vdash x : C, \Delta \qquad \Gamma \vdash x : D, \Delta}{\Gamma \vdash x : C \sqcap D, \Delta}\,(\sqcap) \qquad \frac{\Gamma \vdash x : \exists r.C, y : C, \Delta}{\Gamma \vdash x : \exists r.C, \Delta}\,(\exists r)^{\dagger_2} \qquad \frac{\Gamma, r(x,y) \vdash y : C, y : \dot{\neg}\mathcal{T}_{\mathcal{O}}, \Delta}{\Gamma \vdash x : \forall r.C, \Delta}\,(\forall r)^{\dagger_3}$$

$$\frac{\Gamma' \vdash y_0 : \dot{\neg}C, y_0 : \dot{\neg}\mathcal{T}_{\mathcal{O}}, \ldots, y_n : \dot{\neg}C, y_n : \dot{\neg}\mathcal{T}_{\mathcal{O}}, \Delta}{\Gamma \vdash x : \leqslant nr.C, \Delta}\,(\leqslant nr)^{\dagger_4} \qquad \frac{\Gamma \vdash y_i : C, x : \geqslant nr.C, \Delta \mid 1 \leq i \leq n \qquad \Gamma, y_i \doteq y_j \vdash x : \geqslant nr.C, \Delta \mid 1 \leq i < j \leq n}{\Gamma \vdash x : \geqslant nr.C, \Delta}\,(\geqslant nr)^{\dagger_5}$$

$$\Gamma' = \Gamma, \Gamma^{\not\doteq}(y_0, \ldots, y_n), r(x, y_0), \ldots, r(x, y_n)$$

Side Conditions:

$\dagger_1 =$ '$x =^{*}_{\Gamma} y$.'

$\dagger_2 =$ '$\mathrm{PG}(\Gamma) \vDash [x]_{\Gamma} \overset{L}{\rightsquigarrow} [y]_{\Gamma}$ with $L = L_{\mathrm{G}(\mathcal{O})}(r)$.'

$\dagger_3 =$ 'y is fresh.'

$\dagger_4 =$ 'For each $0 \leq i \leq n$, y_i is fresh.'

$\dagger_5 =$ 'For each $1 \leq i \leq n$, $\mathrm{PG}(\Gamma) \vDash [x]_{\Gamma} \overset{L}{\rightsquigarrow} [y_i]_{\Gamma}$ with $L = L_{\mathrm{G}(\mathcal{O})}(r)$.'

Figure 1: The calculus $\mathsf{S}(\mathcal{O})$ for the $\mathcal{RIQ}$-ontology $\mathcal{O}$. The rules with side conditions $\dagger_1 - \dagger_5$ are applicable only if that side condition holds.

conclusions $S_1, \ldots, S_n$, then applying an n-ary rule (r'), as shown below right, yields a new proof with conclusion S.

$$\frac{}{S}\,(r) \qquad \frac{S_1 \cdots S_n}{S}\,(r')$$

We use π (potentially annotated) to denote proofs, and we say a sequent S is *provable* with π in $\mathsf{S}(\mathcal{O})$, written $\mathsf{S}(\mathcal{O}), \pi \Vdash S$ *iff* S is the conclusion of π. We write $\mathsf{S}(\mathcal{O}) \Vdash S$ to indicate that S is provable with some π in $\mathsf{S}(\mathcal{O})$. Observe that each proof is a tree of sequents with the conclusion as the root. We define the *height* of a proof to be the number of sequents along a maximal branch from the conclusion to an initial rule of the proof. The *size* of a proof π is defined to be the sum of the weights of the sequents it contains; in other words, $s(\pi) := \sum_{S \in \pi} w(S)$, where the *weight* of a sequent $S = \Gamma \vdash \Delta$ is defined to be $w(S) := |\Gamma| + \sum_{x:C \in \Delta} w(C)$. Ignoring labeled concepts of the form $x : \dot{\neg}\mathcal{T}_{\mathcal{O}}$, we refer to the formulae that are explicitly mentioned in the premises of a rule as *active*, and those explicitly mentioned in the conclusion as *principal*. For example, $r(x, y)$ and $y : C$ are active in $(\forall r)$ while $x : \forall r.C$ is principal.

We now define a semantics for our sequents, which is used to establish our sequent systems sound and complete.

Definition 8 (Sequent Semantics)**.** *Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation, $S = \Gamma \vdash \Delta$ a sequent, $\lambda : \mathrm{Lab}(\Gamma, \Delta) \to \Delta^{\mathcal{I}}$ a* label assignment*, and $\mathcal{O}$ an ontology.*

- *$\mathcal{I}, \lambda \vDash^{\forall} \Gamma$ iff for each $r(x, y), x \doteq y, x \not\doteq y \in \Gamma$, we have $(\lambda(x), \lambda(y)) \in r^{\mathcal{I}}$, $\lambda(x) = \lambda(y)$, and $\lambda(x) \neq \lambda(y)$;*
- *$\mathcal{I}, \lambda \vDash^{\exists} \Delta$ iff for some $x : C \in \Delta$, $\lambda(x) \in C^{\mathcal{I}}$.*

A sequent $S = \Gamma \vdash \Delta$ is satisfied *in $\mathcal{I}$ with λ relative to $\mathcal{O}$, written $\mathcal{I}, \lambda \vDash_{\mathcal{O}} S$, iff if $\mathcal{I} \vDash \mathcal{O}$ and $\mathcal{I}, \lambda \vDash^{\forall} \Gamma$, then $\mathcal{I}, \lambda \vDash^{\exists} \Delta$. A sequent $S = \Gamma \vdash \Delta$ is* true *in $\mathcal{I}$ relative to $\mathcal{O}$, written $\mathcal{I} \vDash_{\mathcal{O}} S$, iff $\mathcal{I}, \lambda \vDash_{\mathcal{O}} S$ for all label assignments λ. A sequent $S = \Gamma \vdash \Delta$ is* valid *relative to $\mathcal{O}$, written $\vDash_{\mathcal{O}} S$, iff $\mathcal{I} \vDash_{\mathcal{O}} S$ for all interpretations $\mathcal{I}$, and we say that S is* invalid *relative to $\mathcal{O}$ otherwise, writing $\nvDash_{\mathcal{O}} S$.*

Lemma 2. *Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation, $\mathcal{O}$ be a $\mathcal{RIQ}$ ontology, λ be a label assignment, and Γ be a set of structural atoms. If $\mathcal{I} \vDash \mathcal{O}$, $\mathcal{I}, \lambda \vDash^{\forall} \Gamma$, and $\mathrm{PG}(\Gamma) \vDash [x]_{\Gamma} \overset{L}{\rightsquigarrow} [y]_{\Gamma}$ with $L = L_{\mathrm{G}(\mathcal{O})}(r)$, then $(\lambda(x), \lambda(y)) \in r^{\mathcal{I}}$.*

Theorem 1 (Soundness)**.** *If $\mathsf{S}(\mathcal{O}) \Vdash \Gamma \vdash \Delta$, then $\vDash_{\mathcal{O}} \Gamma \vdash \Delta$.*

We now confirm that $\mathsf{S}(\mathcal{O})$ enjoys desirable proof-theoretic properties, viz. certain rules are height-preserving admissible or invertible. A rule is *(height-preserving) admissible, i.e. (hp-)admissible*, if the premises of the rule have proofs (of heights $h_1, \ldots, h_n$), then the conclusion of the rule has a proof (of height $h \leq \max\{h_1, \ldots, h_n\}$). If we let (r^{-1}) be the inverse of the rule (r) whose premise is the conclusion of (r) and conclusion is the premises of (r), then we say that (r) is *(height-preserving) invertible, i.e. (hp-)invertible iff* (r^{-1}) is (hp-)admissible. For a sequent $S = \Gamma \vdash \Delta$, we let $S(x/y) = \Gamma(x/y) \vdash \Delta(x/y)$ denote the sequent obtained by substituting each occurrence of the label y in S by x; for example, if $S = r(x, y), x \neq y \vdash y : A$, then $S(z/y) = r(x, z), x \neq z \vdash z : A$. Important (hp-)admissible rules are displayed in Figure 2.

Lemma 3. *The $(\top)$ rule is provable in $\mathsf{S}(\mathcal{O})$, and the (ℓ^{x}_{y}), $(w_{\doteq})$, $(w_{\not\doteq})$, (w), (c), and $(s_{\not\doteq})$ rules are hp-admissible.*

Lemma 4. *All non-initial rules in $\mathsf{S}(\mathcal{O})$ are hp-invertible.*

The completeness of $\mathsf{S}(\mathcal{O})$ (stated below) is shown by taking a sequent of the form $\emptyset \vdash x : \dot{\neg}\mathcal{T}_{\mathcal{O}}, x : C$ as input and showing that if the sequent is not provable, then $\mathsf{S}(\mathcal{O})$ can be used to construct a counter-model thereof, witnessing the invalidity of the sequent relative to $\mathcal{O}$.

Theorem 2 (Completeness)**.** *If $\vDash_{\mathcal{O}} \emptyset \vdash x : C$, then $\mathsf{S}(\mathcal{O}) \Vdash \emptyset \vdash x : \dot{\neg}\mathcal{T}_{\mathcal{O}}, x : C$.*

The following corollary is a consequence of Theorem 1 and Theorem 2. We write $\mathsf{S}(\mathcal{O}) \Vdash C \sqsubseteq D$ as shorthand for $\mathsf{S}(\mathcal{O}) \Vdash \emptyset \vdash x : \dot{\neg}\mathcal{T}_{\mathcal{O}}, x : \dot{\neg}C \sqcup D$.

Corollary 1. *$\mathcal{O} \vDash C \sqsubseteq D$ iff $\mathsf{S}(\mathcal{O}) \Vdash C \sqsubseteq D$.*

Last, we emphasize the modularity of our sequent systems and approach. By omitting inference rules for certain connectives and/or only accepting certain ontologies as the input parameter $\mathcal{O}$, sequent calculi can be obtained for expressive and lightweight DLs serving as fragments of $\mathcal{RIQ}$; cf. [Baader *et al.*, 2005; Calvanese and De Giacomo, 2003]. For example, sequent calculi for $\mathcal{ALC}$ ontologies are easily obtained by

$$\frac{}{\Gamma \vdash \Delta, x : \top}\ (\top) \quad \frac{\Gamma \vdash \Delta}{\Gamma(x/y) \vdash \Delta(x/y)}\ (\ell^x_y)^{\dagger_1} \quad \frac{\Gamma \vdash \Delta}{\Gamma, x \doteq y \vdash \Delta}\ (w_{\doteq})^{\dagger_2} \quad \frac{\Gamma \vdash \Delta}{\Gamma, x \not\doteq y \vdash \Delta}\ (w_{\not\doteq})^{\dagger_2}$$

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, x : C}\ (w)^{\dagger_3} \quad \frac{\Gamma \vdash \Delta, x : C, x : C}{\Gamma \vdash \Delta, x : C}\ (c) \quad \frac{\Gamma, x \not\doteq y, y \not\doteq x \vdash \Delta}{\Gamma, x \not\doteq y \vdash \Delta}\ (s_{\not\doteq})$$

Figure 2: (Hp-)admissible rules in $\mathsf{S}(\mathcal{O})$. The side conditions are: $\dagger_1$ = 'x is fresh,' $\dagger_2$ = '$x, y \in \mathrm{Lab}(\Gamma, \Delta)$,' and $\dagger_3$ = '$x \in \mathrm{Lab}(\Gamma, \Delta)$.'

omitting the $(id_{\doteq})$, $(s_{\doteq})$, $(\geqslant nr)$, and $(\leqslant nr)$ rules. The constructive method presented next applies to fragments of $\mathcal{RIQ}$ by leveraging this feature, thus demonstrating its generality.

4 Constructive Sequent-Based Method

We now describe our methodology for computing concept interpolants, and by extension, explicit concept-definitions of implicitly defined concepts (by Lemma 1). The central idea is to generalize the notion of a concept interpolant from GCIs to sequents. Then, given a proof of a sequent S, we assign concept interpolants to all initial sequents of the proof, and show how a concept interpolant can be defined for the conclusion of a rule application from those of its premises, culminating in a concept interpolant for S. As sequents are more general than GCIs, this approach will establish, in a constructive manner, that $\mathcal{RIQ}$ (and its various sublogics) enjoy the concept interpolation property and the CBP.

Definition 9 (Interpolant)**.** *We define an* interpolant *to be a set* $\mathcal{G} := \{\Gamma_i \vdash \Delta_i \mid 1 \leq i \leq n\}$ *such that* Γ_i *is a set of (in)equalities of the form* $x \doteq y$ *and* $x \not\doteq y$ *with* $x, y \in \mathrm{Lab}$ *distinct labels, and* Δ_i *is a set of labeled concepts. Given an interpolant* $\mathcal{G}$ *of the above form, we define its* orthogonal $\overline{\mathcal{G}}$ *as follows:* $\Gamma \vdash \Delta \in \overline{\mathcal{G}}$ *iff for each* $1 \leq i \leq |\Gamma, \Delta|$*, one and only one of the following holds: (1)* $x \doteq y \in \Gamma$ *with* $x \not\doteq y \in \Gamma_i$*, (2)* $x \not\doteq y \in \Gamma$ *with* $x \doteq y \in \Gamma_i$*, or (3)* $x : \dot{\neg} C \in \Delta$ *with* $x : C \in \Delta_i$*. We use* $\mathcal{G}$ *and annotated versions for interpolants.*

Example 2. *Let* $\mathcal{G} = \{(x \doteq y \vdash x : A), (z \not\doteq u \vdash z : \neg B)\}$*. Then, the orthogonal* $\overline{\mathcal{G}}$ *is the set containing* $(x \not\doteq y, z \doteq u \vdash)$, $(x \not\doteq y \vdash z : B)$, $(z \doteq u \vdash x : \neg A)$*, and* $(\ \vdash x : \neg A, z : B)$*, that is, each member of* $\overline{\mathcal{G}}$ *is formed by including a negated element from each member of* $\mathcal{G}$*.*

In order to fully specify our interpolant construction algorithm, we need to define two special interpolants, named $\forall r.\mathcal{G}$ and $\leqslant nr.\mathcal{G}$, which appear in quantifier and qualified number restriction rules. We let $\vec{C}$ denote a set of complex concepts, define $x : \vec{C} := \{x : C \mid C \in \vec{C}\}$, and define $\bigsqcap \vec{C}$, $\bigsqcup \vec{C}$, and $\dot{\neg}\vec{C}$ to be the conjunction, disjunction, and negation of all complex concepts in $\vec{C}$, respectively.

Definition 10. *Let* $\mathcal{G} = \{\Gamma \vdash \Delta_i, y : \vec{C}_i \mid 1 \leq i \leq m\}$ *such that* $y \notin \mathrm{Lab}(\Gamma)$ *and* $\Delta_i \restriction y = \emptyset$*, then we define:*

$$\forall r.\mathcal{G} := \{\Gamma \vdash \Delta_i, x : \forall r. \bigsqcup \vec{C}_i \mid 1 \leq i \leq m\}.$$

Let $\mathcal{G} = \{\Gamma, \Gamma' \vdash \Delta_i, y_0 : \vec{C}_{0,i}, \ldots, y_n : \vec{C}_{n,i} \mid 1 \leq i \leq m\}$ *such that* $\mathrm{Lab}(\Gamma) \cap \{y_0, \ldots, y_n\} = \emptyset$, $\Gamma' \subseteq \Gamma^{\not\doteq}(y_0, \ldots, y_n)$, $\Delta_i \restriction y_j = \emptyset$ *for* $0 \leq j \leq n$*, and* $\vec{C}_i = \vec{C}_{0,i}, \ldots, \vec{C}_{n,i}$*. Then,*

$$\leqslant nr.\mathcal{G} := \{(\Gamma \vdash \Delta_i, x : \leqslant nr. \dot{\neg} \bigsqcup \vec{C}_i) \mid 1 \leq i \leq m\}.$$

An *interpolation sequent* is defined to be an expression of the form $\Gamma; \Phi\ {}^a|^b\ \Psi \vdash \Delta\ {}^a|^b\ \Sigma \parallel \mathcal{G}$ such that Γ is a set of role and equality atoms, Φ, Ψ is a set of inequality atoms, Δ, Σ is a multiset of labeled concepts, $\mathcal{G}$ is an interpolant, and $a, b \in \{1, 2\}$ with $a \neq b$. For an interpolation sequent of the aforementioned form, we refer to $\Gamma, \Phi \vdash \Delta$ as the *left partition* and $\Gamma, \Psi \vdash \Sigma$ as the *right partition*. Recall that for a concept interpolant I of a GCI $C \sqsubseteq D$ under $\mathcal{O}$, the ontology $\mathcal{O}$ is the union of two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ such that $\mathrm{con}(I) \subseteq \mathrm{con}(\mathcal{O}_1, C) \cap \mathrm{con}(\mathcal{O}_2, D)$ (see Definition 3). The use of $a, b \in \{1, 2\}$ in an interpolation sequent is to keep track of which partition is associated with which ontology, e.g. in $\Gamma; \Phi\ {}^1|^2\ \Psi \vdash \Delta\ {}^1|^2\ \Sigma \parallel \mathcal{G}$ the left (right) partition is associated with $\mathcal{O}_1$ ($\mathcal{O}_2$, respectively).

Definition 11 (Interpolant Preserving Rules)**.** *Let* (r) *be a rule in the set* $\{(s_{\doteq}), (\sqcup), (\sqcap), (\exists r), (\geqslant nr)\}$ *of the form shown below and assume that the active equalities and/or labeled concepts occur in* Γ_i *and/or* Σ_i*, respectively, with the principal formula in* Σ*.*

$$\frac{\Gamma_i, \Phi, \Psi \vdash \Delta, \Sigma_i \mid 1 \leq i \leq n}{\Gamma, \Phi, \Psi \vdash \Delta, \Sigma}\ (r)$$

We define its corresponding interpolant rule *as follows:*

$$\frac{\Gamma_i; \Phi\ {}^a|^b\ \Psi \vdash \Delta\ {}^a|^b\ \Sigma_i \parallel \mathcal{G}_i \mid 1 \leq i \leq n}{\Gamma; \Phi\ {}^a|^b\ \Psi \vdash \Delta\ {}^a|^b\ \Sigma \parallel \mathcal{G}_1 \cup \cdots \cup \mathcal{G}_n}\ (r^I)$$

We refer to a rule $(s^I_{\doteq})$, $(\sqcup^I)$, $(\sqcap^I)$, $(\exists r^I)$*, or* $(\geqslant nr^I)$ *as an* interpolant preserving rule*, or* IP-rule*. We stipulate that* $(\exists r^I)$ *and* $(\geqslant nr^I)$ *are subject to the same side conditions as* $(\exists r)$ *and* $(\geqslant nr)$*, respectively, w.r.t. the propagation graph* $\mathrm{PG}(\Gamma)$.

For each sequent calculus $\mathsf{S}(\mathcal{O})$, we define its corresponding *interpolation calculus* accordingly:

$$\mathsf{SI}(\mathcal{O}) := \{(id^I_1), (id^I_2), (O)\} \cup \{(r^I) \mid (r) \in \mathsf{S}(\mathcal{O}) \setminus (id)\}$$

Observe that interpolation calculi contain IP-rules as well as rules from Figure 3. In an interpolation calculus $\mathsf{SI}(\mathcal{O})$, the (id^I_1), (id^I_2), and $(id^I_{\doteq})$ rules are the *initial rules*, (O) is the *orthogonal rule*, $(s^I_{\doteq})$ is the *substitution rule*, and all remaining rules are *logical rules*. The orthogonal rule cuts the number of rules needed in $\mathsf{SI}(\mathcal{O})$ roughly in half as it essentially 'swaps' the left and right partition permitting rules to be defined that only operate within the right partition; cf. [Lyon *et al.*, 2020]. A *proof*, its *height*, and the provability relation $\Vdash$ are defined in $\mathsf{SI}(\mathcal{O})$ in the same manner as for $\mathsf{S}(\mathcal{O})$.

We now put forth a sequence of lemmas culminating in the main interpolation theorem (Theorem 3), which implies that $\mathcal{RIQ}$ has the CBP (Corollary 2). We remark that Lemmas 5 and 7 describe proof transformation algorithms between $\mathsf{S}(\mathcal{O})$ and $\mathsf{SI}(\mathcal{O})$. In particular, Lemma 5 states that

$$\frac{}{\Gamma;\Phi\ {}^{a}|^{b}\ \Psi\vdash\Delta,x:A\ {}^{a}|^{b}\ x:\neg A,\Sigma\parallel\{(\ \vdash x:\neg A)\}}\ (id^I_1)\qquad\frac{}{\Gamma;\Phi\ {}^{a}|^{b}\ \Psi\vdash\Delta\ {}^{a}|^{b}\ x:A,x:\neg A,\Sigma\parallel\{(\ \vdash x:\top)\}}\ (id^I_2)$$

$$\frac{}{\Gamma;\Phi\ {}^{a}|^{b}\ \Psi,x\not\doteq y\vdash\Delta\ {}^{a}|^{b}\ \Sigma\parallel\{(x\not\doteq y\vdash\ )\}}\ (id^I_{\doteq})$$

$$\frac{\Gamma;\Phi\ {}^{a}|^{b}\ \Psi\vdash\Delta\ {}^{a}|^{b}\ \Sigma\parallel\mathcal{G}}{\Gamma;\Psi\ {}^{b}|^{a}\ \Phi\vdash\Sigma\ {}^{b}|^{a}\ \Delta\parallel\overline{\mathcal{G}}}\ (O)\qquad\frac{\Gamma,r(x,y);\Phi\ {}^{a}|^{b}\ \Psi\vdash\Delta\ {}^{a}|^{b}\ y:C,y:\dot\neg\mathcal{T}_{\mathcal{O}},\Sigma\parallel\mathcal{G}}{\Gamma;\Phi\ {}^{a}|^{b}\ \Psi\vdash\Delta\ {}^{a}|^{b}\ x:\forall r.C,\Sigma\parallel\forall r.\mathcal{G}}\ (\forall r^I)$$

$$\frac{\Gamma,r(x,y_0),\ldots,r(x,y_n);\Phi\ {}^{a}|^{b}\ \Psi,\Gamma^{\not\doteq}(y_0,\ldots,y_n)\vdash\Delta\ {}^{a}|^{b}\ y_0:\dot\neg C,y_0:\dot\neg\mathcal{T}_{\mathcal{O}},\ldots,y_n:\dot\neg C,y_n:\dot\neg\mathcal{T}_{\mathcal{O}},\Sigma\parallel\mathcal{G}}{\Gamma;\Phi\ {}^{a}|^{b}\ \Psi\vdash\Delta\ {}^{a}|^{b}\ x:\leqslant nr.C,\Sigma\parallel\leqslant nr.\mathcal{G}}\ (\leqslant nr^I)$$

Figure 3: Rules in $\mathsf{SI}(\mathcal{O})$. The $(id^I_{\doteq})$, $(\forall r^I)$, and $(\leqslant nr^I)$ rules satisfy the same side conditions as $(id_{\doteq})$, $(\forall r)$, and $(\leqslant nr)$, respectively.

each proof in $\mathsf{S}(\mathcal{O})$ of a sequent $\Gamma,\Phi,\Psi\vdash\Delta,\Sigma$ in a special form can be transformed into a proof in $\mathsf{SI}(\mathcal{O})$ of a specific interpolation sequent $\Gamma;\Phi\ {}^{a}|^{b}\ \Psi\vdash\Delta\ {}^{a}|^{b}\ \Sigma\parallel\mathcal{G}$. Then, via Lemma 7, this proof can be transformed into two proofs in $\mathsf{S}(\mathcal{O})$ witnessing that the interpolant $\mathcal{G}$ is 'implied by' the left partition $\Gamma,\Phi\vdash\Delta$ and 'implies' the right partition $\Gamma,\Psi\vdash\Sigma$. Both Lemmas 5 and 7 are shown by induction on the height of the given proof. Last, when we use the notation $\Gamma,\Phi_a,\Phi_b\vdash\dot\neg\mathcal{T}_{\mathcal{O}_a},\Delta_a,\Delta_b,\dot\neg\mathcal{T}_{\mathcal{O}_b}$ or the notation $\Gamma;\Phi_a\ {}^{a}|^{b}\ \Phi_b\vdash\dot\neg\mathcal{T}_{\mathcal{O}_a},\Delta_a\ {}^{a}|^{b}\ \Delta_b,\dot\neg\mathcal{T}_{\mathcal{O}_b}\parallel\mathcal{G}$, we assume that $\dot\neg\mathcal{T}_{\mathcal{O}_c}:=x_1:\dot\neg\mathcal{T}_{\mathcal{O}_c},\ldots,x_n:\dot\neg\mathcal{T}_{\mathcal{O}_c}$ such that $\mathrm{Lab}(\Gamma,\Phi_c,\Delta_c)=\{x_1,\ldots,x_n\}$ and $c\in\{a,b\}$. The use of $\dot\neg\mathcal{T}_{\mathcal{O}_c}$ ensures each partition satisfies its respective ontology.

Lemma 5. *Let $\mathcal{O}=\mathcal{O}_1\cup\mathcal{O}_2$ be an ontology and suppose that $\Gamma,\Phi,\Psi\vdash\dot\neg\mathcal{T}_{\mathcal{O}_a},\Delta,\Sigma,\dot\neg\mathcal{T}_{\mathcal{O}_b}$ has a proof π in $\mathsf{S}(\mathcal{O})$ with $\Phi\cap\Psi=\emptyset$. Then, π can be transformed into a proof in $\mathsf{SI}(\mathcal{O})$ of $\Gamma;\Phi\ {}^{a}|^{b}\ \Psi\vdash\dot\neg\mathcal{T}_{\mathcal{O}_a},\Delta\ {}^{a}|^{b}\ \Sigma,\dot\neg\mathcal{T}_{\mathcal{O}_b}\parallel\mathcal{G}$ such that:*

(1) If $x\doteq y$ occurs in $\mathcal{G}$, then $x\not\doteq y\in\Phi$;

(2) If $x\not\doteq y$ occurs in $\mathcal{G}$, then $x\not\doteq y\in\Psi$;

(3) $\mathrm{Lab}(\mathcal{G})\subseteq\mathrm{Lab}(\Gamma,\Phi,\Psi,\dot\neg\mathcal{T}_{\mathcal{O}_a},\Delta,\Sigma,\dot\neg\mathcal{T}_{\mathcal{O}_b})$;

(4) $\mathrm{con}(\mathcal{G})\subseteq\mathrm{con}(\mathcal{O}_a,\Delta)\cap\mathrm{con}(\Sigma,\mathcal{O}_b)$.

The following lemma states that a double orthogonal transformation always 'preserves' some of the sequents from the original interpolant. As proven in the appended version [Lyon and Karge, 2024], the lemma is helpful in showing Lemma 7.

Lemma 6. *If $(\Sigma\vdash\Pi)\in\overline{\overline{\mathcal{G}}}$, then there exists a $(\Gamma\vdash\Delta)\in\mathcal{G}$ such that $\Gamma\subseteq\Sigma$ and $\Delta\subseteq\Pi$.*

Lemma 7. *If $\mathsf{SI}(\mathcal{O})\Vdash\Gamma;\Phi\ {}^{a}|^{b}\ \Psi\vdash\Delta\ {}^{a}|^{b}\ \Sigma\parallel\mathcal{G}$, then*

(1) For each $(\Gamma'\vdash\Pi_i)\in\mathcal{G}$, $\mathsf{S}(\mathcal{O})\Vdash\Gamma,\Gamma',\Phi\vdash\Delta,\Pi_i$;

(2) For each $(\Gamma'\vdash\Pi_i)\in\overline{\mathcal{G}}$, $\mathsf{S}(\mathcal{O})\Vdash\Gamma,\Gamma',\Psi\vdash\Pi_i,\Sigma$.

Next, we prove that an interpolant containing at most a single label, i.e. an interpolant of the form

$$\mathcal{G}:=\{(\ \vdash x:C_{i,1},\ldots,x:C_{i,k_i})\mid 1\leq i\leq n\}$$

can be transformed into a single labeled concept within the context of a proof. Toward this end, we define $x:\bigsqcap\bigsqcup\mathcal{G}:=x:\bigsqcap_{1\leq i\leq n}\bigsqcup_{1\leq j\leq k_i}C_{i,j}$, where $\mathcal{G}$ is as above. The following two lemmas are straightforward and follow by applying the $(\sqcup)$ and $(\sqcap)$ rules in $\mathsf{S}(\mathcal{O})$ a sufficient number of times.

Lemma 8. *If $\Gamma\vdash\Delta,\Sigma$ is provable in $\mathsf{S}(\mathcal{O})$ for all $(\ \vdash\Sigma)\in\mathcal{G}$ and $\mathrm{Lab}(\mathcal{G})=\{x\}$, then $\mathsf{S}(\mathcal{O})\Vdash\Gamma\vdash\Delta,x:\bigsqcap\bigsqcup\mathcal{G}$.*

Lemma 9. *If $\Gamma\vdash\Delta,\Sigma$ is provable in $\mathsf{S}(\mathcal{O})$ for all $(\ \vdash\Sigma)\in\overline{\mathcal{G}}$ and $\mathrm{Lab}(\mathcal{G})=\{x\}$, then $\mathsf{S}(\mathcal{O})\Vdash\Gamma\vdash\Delta,x:\dot\neg\bigsqcap\bigsqcup\mathcal{G}$.*

Our main theorem below is a consequence of Lemmas 4–9. Given a proof of $\vdash x:\dot\neg\mathcal{T}_{\mathcal{O}},x:\dot\neg C\sqcup D$, we obtain proofs of $\vdash x:\dot\neg\mathcal{T}_{\mathcal{O}_1},x:\dot\neg C,x:I$ and $\vdash x:\dot\neg\mathcal{T}_{\mathcal{O}_2},x:D,x:\dot\neg I$ in $\mathsf{S}(\mathcal{O})$ by Lemmas 4, 5, and 7–9 with $I=\bigsqcap\bigsqcup\mathcal{G}$. The concept interpolant I is computed in $\mathsf{EXPTIME}$ due to the potential use of the (O) rule, which may exponentially increase the size of interpolants.

Theorem 3. *Let $\mathcal{O}=\mathcal{O}_1\cup\mathcal{O}_2$ be a $\mathcal{RIQ}$ ontology. If $\mathcal{O}\vDash C\sqsubseteq D$, i.e. $\mathsf{S}(\mathcal{O}),\pi\Vdash C\sqsubseteq D$, then a concept interpolant I can be computed in $\mathsf{EXPTIME}$ relative to $s(\pi)$ such that $\mathsf{S}(\mathcal{O})\Vdash C\sqsubseteq I$ and $\mathsf{S}(\mathcal{O})\Vdash I\sqsubseteq D$, i.e. $\mathcal{O}\vDash C\sqsubseteq I$ and $\mathcal{O}\vDash I\sqsubseteq D$.*

Let C be a complex concept, $\mathcal{O}$ be a $\mathcal{RIQ}$ ontology, $\Theta\subseteq\mathrm{con}(C,\mathcal{O})$, and suppose C is implicitly concept-definable from Θ under $\mathcal{O}$. If we want to find the explicit concept-definition of C from Θ under $\mathcal{O}$, we utilize the sequent calculus $\mathsf{S}(\mathcal{O}')$ with $\mathcal{O}'=\mathcal{O}\cup\mathcal{O}_\Theta$. Since C is implicitly concept-definable, we know by (1) in Section 2.2 and Corollary 1 that $\mathsf{S}(\mathcal{O}')\Vdash C\sqsubseteq C_\Theta$. By applying Theorem 3, we obtain a concept interpolant I for $C\sqsubseteq C_\Theta$ under $\mathcal{O}$, which serves as an explicit concept-definition by Lemma 1. Therefore, we have a constructive proof of the following corollary.

Corollary 2. *$\mathcal{RIQ}$ has the concept interpolation property and the CBP.*

5 Concluding Remarks

There are various avenues of future research. First, it would be interesting to know the size and complexity of computing a concept interpolant I relative to $C\sqsubseteq D$ rather than from a proof π witnessing $\mathcal{O}\vDash C\sqsubseteq D$. This can be achieved by supplying a proof-search algorithm that generates a proof of $C\sqsubseteq D$, whose relative complexity and size can then be determined. Second, we aim to generalize our methodology to decide and compute the existence of *Craig interpolants* for $\mathcal{RIQ}$ and related DLs, which is a non-trivial problem (see [ten Cate *et al.*, 2013]). Last, we could generalize our method to consider constructs beyond those in $\mathcal{RIQ}$, e.g. negations over roles, intersections of roles, nominals, or the @ operator; it is known that for some of these extensions, e.g. nominals, even concept interpolation fails [Artale *et al.*, 2023], requiring an increase in complexity to decide the existence of interpolants.

Acknowledgments

Tim S. Lyon is supported by the European Research Council (ERC) Consolidator Grant 771779 (DeciGUT). Jonas Karge is supported by BMBF (Federal Ministry of Education and Research) in DAAD project 57616814 (SECAI, School of Embedded Composite AI) as well as by BMBF in the Center for Scalable Data Analytics and Artificial Intelligence (ScaDS.AI).

References

[Areces *et al.*, 2008] Carlos Areces, Alexander Koller, and Kristina Striegnitz. Referring expressions as formulas of description logic. In *Proceedings of the Fifth International Natural Language Generation Conference*, INLG'08, page 42–49, USA, 2008. Association for Computational Linguistics.

[Artale *et al.*, 2021] Alessandro Artale, Andrea Mazzullo, Ana Ozaki, and Frank Wolter. On free description logics with definite descriptions. In Meghyn Bienvenu, Gerhard Lakemeyer, and Esra Erdem, editors, *Proceedings of the 18th International Conference on Principles of Knowledge Representation and Reasoning, KR 2021, Online event, November 3-12, 2021*, pages 63–73, 2021.

[Artale *et al.*, 2023] Alessandro Artale, Jean Christoph Jung, Andrea Mazzullo, Ana Ozaki, and Frank Wolter. Living without beth and craig: Definitions and interpolants in description and modal logics with nominals and role inclusions. *ACM Transactions on Computational Logic*, 24(4):1–51, 2023.

[Baader and Nutt, 2003] Franz Baader and Werner Nutt. *Basic Description Logics*, page 43–95. Cambridge University Press, USA, 2003.

[Baader *et al.*, 2005] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the EL envelope. In Leslie Pack Kaelbling and Alessandro Saffiotti, editors, *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence*, pages 364–369. Professional Book Center, 2005.

[Beth, 1956] Evert Willem Beth. On padoa's method in the theory of definition. *Journal of Symbolic Logic*, 21(2):194–195, 1956.

[Borgida *et al.*, 2016] Alexander Borgida, David Toman, and Grant E. Weddell. On referring expressions in query answering over first order knowledge bases. In Chitta Baral, James P. Delgrande, and Frank Wolter, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Fifteenth International Conference, KR 2016*, pages 319–328. AAAI Press, 2016.

[Calvanese and De Giacomo, 2003] Diego Calvanese and Giuseppe De Giacomo. *Expressive Description Logics*, page 178–218. Cambridge University Press, USA, 2003.

[Castilho *et al.*, 1997] Marcos A Castilho, Luis Farinas del Cerro, Olivier Gasquet, and Andreas Herzig. Modal tableaux with propagation rules and structural rules. *Fundamenta Informaticae*, 32(3, 4):281–297, 1997.

[Ciabattoni *et al.*, 2021] Agata Ciabattoni, Tim S. Lyon, Revantha Ramanayake, and Alwen Tiu. Display to labeled proofs and back again for tense logics. *ACM Trans. Comput. Logic*, 22(3), 2021.

[Craig, 1957] William Craig. Three uses of the herbrand-gentzen theorem in relating model theory and proof theory. *The Journal of Symbolic Logic*, 22(3):269–285, 1957.

[Donini *et al.*, 1997] Francesco M Donini, Maurizio Lenzerini, Daniele Nardi, and Werner Nutt. The complexity of concept languages. *Information and Computation*, 134(1):1–58, 1997.

[Donini, 2003] Francesco M. Donini. *Complexity of Reasoning*, page 96–136. Cambridge University Press, USA, 2003.

[Dyckhoff, 1992] Roy Dyckhoff. Contraction-free sequent calculi for intuitionistic logic. *The Journal of Symbolic Logic*, 57(3):795–807, 1992.

[Fitting and Kuznets, 2015] Melvin Fitting and Roman Kuznets. Modal interpolation via nested sequents. *Annals of pure and applied logic*, 166(3):274–305, 2015.

[Fitting, 1972] Melvin Fitting. Tableau methods of proof for modal logics. *Notre Dame Journal of Formal Logic*, 13(2):237–247, 1972.

[Franconi and Kerhet, 2019] Enrico Franconi and Volha Kerhet. Effective query answering with ontologies and dboxes. In Carsten Lutz, Uli Sattler, Cesare Tinelli, Anni-Yasmin Turhan, and Frank Wolter, editors, *Description Logic, Theory Combination, and All That: Essays Dedicated to Franz Baader on the Occasion of His 60th Birthday*, pages 301–328. Springer International Publishing, Cham, 2019.

[Funk *et al.*, 2019] Maurice Funk, Jean Christoph Jung, Carsten Lutz, Hadrien Pulcini, and Frank Wolter. Learning description logic concepts: When can positive and negative examples be separated? In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019*, pages 1682–1688. ijcai.org, 2019.

[Gentzen, 1935a] Gerhard Gentzen. Untersuchungen über das logische schließen. i. *Mathematische zeitschrift*, 39(1):176–210, 1935.

[Gentzen, 1935b] Gerhard Gentzen. Untersuchungen über das logische schließen. ii. *Mathematische Zeitschrift*, 39(1):405–431, 1935.

[Goré *et al.*, 2011] Rajeev Goré, Linda Postniece, and Alwen Tiu. On the correspondence between display postulates and deep inference in nested sequent calculi for tense logics. *Log. Methods Comput. Sci.*, 7(2), 2011.

[Horrocks and Sattler, 2004] Ian Horrocks and Ulrike Sattler. Decidability of shiq with complex role inclusion axioms. *Artificial Intelligence*, 160(1-2):79–104, 2004.

[Jung *et al.*, 2022] Jean Christoph Jung, Andrea Mazzullo, and Frank Wolter. More on interpolants and explicit definitions for description logics with nominals and/or role inclusions. *Proceedings of DL2022*, 2022.

[Kuznets and Lellmann, 2018] Roman Kuznets and Björn Lellmann. Interpolation for intermediate logics via hyper- and linear nested sequents. In Guram Bezhanishvili, Giovanna D'Agostino, George Metcalfe, and Thomas Studer, editors, *Advances in Modal Logic, Proceedings*, pages 473–492. College Publications, 2018.

[Lyon and Karge, 2024] Tim S. Lyon and Jonas Karge. Constructive interpolation and concept-based beth definability for description logics via sequents. https://arxiv.org/abs/2404.15840, 2024.

[Lyon *et al.*, 2020] Tim Lyon, Alwen Tiu, Rajeev Goré, and Ranald Clouston. Syntactic interpolation for tense logics and bi-intuitionistic logic via nested sequents. In Maribel Fernández and Anca Muscholl, editors, *28th EACSL Annual Conference on Computer Science Logic, CSL 2020*, volume 152 of *LIPIcs*, pages 28:1–28:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.

[Lyon, 2021] Tim S. Lyon. Nested sequents for intuitionistic modal logics via structural refinement. In Anupam Das and Sara Negri, editors, *Automated Reasoning with Analytic Tableaux and Related Methods*, pages 409–427, Cham, 2021. Springer International Publishing.

[Maehara, 1960] Shoji Maehara. On the interpolation theorem of craig. *Sûgaku*, 12(4):235–237, 1960.

[Post, 1947] Emil L Post. Recursive unsolvability of a problem of thue. *The Journal of Symbolic Logic*, 12(1):1–11, 1947.

[Seylan *et al.*, 2009] Inanç Seylan, Enrico Franconi, and Jos de Bruijn. Effective query rewriting with ontologies over dboxes. In Craig Boutilier, editor, *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence*, pages 923–925, 2009.

[Slaney, 1997] John K. Slaney. Minlog: A minimal logic theorem prover. In William McCune, editor, *Automated Deduction - CADE-14, 14th International Conference on Automated Deduction, Proceedings*, volume 1249 of *Lecture Notes in Computer Science*, pages 268–271. Springer, 1997.

[ten Cate *et al.*, 2006] Balder ten Cate, Willem Conradie, Maarten Marx, and Yde Venema. Definitorially complete description logics. In Patrick Doherty, John Mylopoulos, and Christopher A. Welty, editors, *Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning*, pages 79–89. AAAI Press, 2006.

[ten Cate *et al.*, 2013] Balder ten Cate, Enrico Franconi, and Inanç Seylan. Beth definability in expressive description logics. *Journal of Artificial Intelligence Research*, 48:347–414, 2013.

[Toman and Weddell, 2022] David Toman and Grant E. Weddell. First order rewritability in ontology-mediated querying in horn description logics. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022*, pages 5897–5905. AAAI Press, 2022.