Practical Uniform Interpolation and Forgetting for $\mathcal{ALC}$ TBoxes with Applications to Logical Difference

Michel Ludwig
Institute for Theoretical Computer Science
TU Dresden, Germany

Boris Konev
Department of Computer Science
University of Liverpool, United Kingdom

Abstract

We develop a clausal resolution-based approach for computing uniform interpolants of TBoxes formulated in the description logic $\mathcal{ALC}$ when such uniform interpolants exist. We also present an experimental evaluation of our approach and of its application to the logical difference problem for real-life $\mathcal{ALC}$ ontologies. Our results indicate that in many practical cases uniform interpolants exist and that they can be computed with the presented algorithm.

Introduction

Ontologies or *TBoxes* expressed in Description Logics (DL) provide a common vocabulary for a domain of interest together with a description of the meaning of the terms built from the vocabulary and of the relationships between them. Modern applications of ontologies, especially in the biological, medical, or healthcare domain, often demand large and complex ontologies; for example, the National Cancer Institute ontology (NCI) consists of more than 60 000 term definitions. For developing, maintaining, and deploying such large-scale ontologies it can be advantageous for ontology engineers to concentrate on specific parts of an ontology and ignore or *forget* the rest. Ignoring parts of an ontology can be formalised with the help of *predicate forgetting* and its dual *uniform interpolation*, which have both been extensively studied in the AI and DL literature (ten Cate et al. 2006; Eiter et al. 2006; Herzig and Mengin 2008; Konev, Walther, and Wolter 2009; Wang et al. 2008; 2010; Lutz and Wolter 2011; Wang et al. 2012).

Forgetting parts of an ontology can be used, for example, in the following practical scenarios. *Exhibiting hidden relations*: in addition to the explicitly stated connections between terms, additional relations can also be derived from ontologies with the help of reasoners. Such inferred connections are often harder to understand or debug. By forgetting everything but a handful of terms of interest, it then becomes possible to exhibit inferred relations that were hidden initially, potentially simplifying the understanding of the ontology structure. *Ontology obfuscation*: in software engineering, obfuscation (Collberg, Thomborson, and Low 1998) transforms a given program into a functionally equivalent one that is more difficult to read and understand for humans for the purpose of preventing reverse engineering. Forgetting can provide a similar function in the context of ontology engineering. Terms are often defined with the help of auxiliary terms which give structure to TBox inclusions. However, such a structure might be considered proprietary knowledge that should not be exposed, or it could simply be of little interest for ontology users. By forgetting these intermediate auxiliary terms, we obtain an ontology that is functionally equivalent, yet harder to read, understand, and modify by humans. Further applications of forgetting can be found in (Konev, Walther, and Wolter 2009; Lutz, Seylan, and Wolter 2012).

A promising and important application area of forgetting is the computation of the *logical difference* between ontology versions. Determining whether two versions of a document have differences is a standard task in information technology, and finding differences is particularly relevant for text processing and software development. Already in these areas, it is important to be able to identify which changes are *significant* and which are not (e.g., a software developer might want to ignore changes in the formatting style of the code such as the number of indentation spaces). Detecting significant changes is even more important in the setting of Knowledge Representation, where differences in the knowledge captured by ontologies are often more relevant than syntactic changes. Arguably, one of the most important concerns of an ontology engineer when modifying an existing ontology is to ensure that the introduced changes do not interfere with the meaning of the terms outside the fragment under consideration. Notice that neither the version comparison based on the syntactic form of the documents representing ontologies (Conradi and Westfechtel 1998) nor methods based on the structural transformations of ontology statements (Noy and Musen 2002; Klein et al. 2002; Jiménez-Ruiz et al. 2011) can be used to identify changes to the logical meaning of terms in every situation. However, such a correctness guarantee can be achieved by checking the equivalence of the ontologies resulting from forgetting the terms under consideration before and after the changes occurred.

In this paper we develop an algorithm based on clausal resolution for computing uniform interpolants of TBoxes formulated in the description logic $\mathcal{ALC}$ which can preserve all the consequences that do not make use of some given *concept names*. Subsequently, we present an experimental evaluation of our approach which demonstrates that in many practical cases uniform interpolants exist and that they can be computed with our algorithm. We also apply our prototype tool to compute the logical difference between versions of ontologies from the biomedical domain.

This is an updated and extended version of (Ludwig and Konev 2013). All missing proofs can be found in the full version of this paper, which is available from `http://lat.inf.tu-dresden.de/~michel/kr14-ui-full.pdf`

Related work. Until recently research on uniform interpolation and forgetting in the setting of DL mainly has concentrated on theoretical foundations of forgetting. This could be partly explained by the high computational complexity of this task and by the fact that uniform interpolants do not always exist. The notion of forgetting has been introduced by Reiter and Lin (1994). (Konev, Walther, and Wolter 2009) prove tractability of uniform interpolation for $\mathcal{EL}$ TBoxes of a specific syntactic form. (Wang et al. 2008; 2010; 2012) have developed algorithms for forgetting in expressive description logics. A tight 2-EXPTIME-complete bound on the complexity for deciding the existence of a Σ-uniform interpolant in $\mathcal{ALC}$ and a worst-case triple-exponential procedure for computing a Σ-uniform interpolant if it exists, have been given in (Lutz and Wolter 2011). Koopmann and Schmidt (2013) have introduced a two-stage resolution-based algorithm for computing uniform interpolants. As outcome of the first stage, a representation of the uniform interpolant in a description logic with fixpoint operators is computed (such a representation always exists) Then in the second stage an attempt is made to eliminate the newly-introduced fixpoints (which may not succeed). In contrast to this approach, our algorithm has one stage and it can be guaranteed that a uniform interpolant of bounded depth is returned.

The notion of the logical difference has been introduced in (Konev, Walther, and Wolter 2008) as a way of capturing the difference in the meaning of terms that is independent of the representation of ontologies.

Preliminaries

We start with introducing the description logic $\mathcal{ALC}$. Let $\mathsf{N_C}$ and $\mathsf{N_R}$ be countably infinite and mutually disjoint sets of *concept names* and *role names*. *$\mathcal{ALC}$-concepts* are built according to the following syntax rule

$$C ::= \quad A \quad | \quad \top \quad | \quad \neg C \quad | \quad \exists r.C \quad | \quad C \sqcap D,$$

where $A \in \mathsf{N_C}$ and $r \in \mathsf{N_R}$. As usual, other $\mathcal{ALC}$ concept constructors are introduced as abbreviations: $\bot$ stands for $\neg\top$, $C \sqcup D$ stands for $\neg(\neg C \sqcap \neg D)$ and $\forall r.C$ stands for $\neg\exists r.\neg C$. An $\mathcal{ALC}$-TBox $\mathcal{T}$ is a finite set of $\mathcal{ALC}$-inclusions of the form $C \sqsubseteq D$, where C and D are $\mathcal{ALC}$-concepts. A concept equation $C \equiv D$ is an abbreviation for the two inclusions $C \sqsubseteq D$ and $D \sqsubseteq C$. An $\mathcal{ALC}$-TBox is *acyclic* if all its inclusions are of the form $A \sqsubseteq C$ and $A \equiv C$, where $A \in \mathsf{N_C}$ and C is an $\mathcal{ALC}$-concept, such that no concept name occurs more than once on the left-hand side and $\mathcal{T}$ contains no cycle in its definitions, that is, it does not contain inclusions $A_1 \bowtie C_1, \dots, A_k \bowtie C_k$, where $\bowtie \in \{\sqsubseteq, \equiv\}$, such that A_{i+1} occurs in C_i, for $i = 1, \dots, k-1$ and $A_k = A_1$.

A signature Σ is a finite subset of $\mathsf{N_C} \cup \mathsf{N_R}$. The signature of a concept C, denoted by $\mathsf{sig}(C)$, is the set of concept and role names that occur in C. If $\mathsf{sig}(C) \subseteq \Sigma$, we call C a Σ-concept. We assume that the two previous definitions also apply to concept inclusions/equations $C \bowtie D$ with $\bowtie \in \{\sqsubseteq, \equiv\}$ and to TBoxes $\mathcal{T}$. The size of a concept C is the length of the string that represents it, where concept names and role names are considered to be of length one. The size of an inclusion/equation $C \bowtie D$ with $\bowtie \in \{\sqsubseteq, \equiv\}$ is the sum of the sizes of C and D plus one. The size of a TBox $\mathcal{T}$ is the sum of the sizes of its inclusions.

The semantics of $\mathcal{ALC}$ is given by *interpretations* $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where the *domain* $\Delta^\mathcal{I}$ is a non-empty set, and $\cdot^\mathcal{I}$ is a function mapping each concept name A to a subset $A^\mathcal{I}$ of $\Delta^\mathcal{I}$, each role name r to a binary relation $r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$. The *extension* $C^\mathcal{I}$ of a concept C is defined by induction as follows:

$$\begin{array}{lcl} \top^\mathcal{I} & := & \Delta^\mathcal{I} \\ (\neg C)^\mathcal{I} & := & \Delta^\mathcal{I} \setminus C^\mathcal{I} \\ (\exists r.C)^\mathcal{I} & := & \{\, d \in \Delta^\mathcal{I} \mid \exists e \in C^\mathcal{I} : (d,e) \in r^\mathcal{I} \,\} \\ (C \sqcap D)^\mathcal{I} & := & C^\mathcal{I} \cap D^\mathcal{I}. \end{array}$$

Then $\mathcal{I}$ *satisfies* a concept inclusion $C \sqsubseteq D$, in symbols $\mathcal{I} \models C \sqsubseteq D$, if $C^\mathcal{I} \subseteq D^\mathcal{I}$.

We say that an interpretation $\mathcal{I}$ is a *model of a TBox* $\mathcal{T}$ if $\mathcal{I} \models C \sqsubseteq D$ for all $C \sqsubseteq D \in \mathcal{T}$. An $\mathcal{ALC}$-inclusion $C \sqsubseteq D$ *follows from* (or is *entailed by*) a TBox $\mathcal{T}$ if every model of $\mathcal{T}$ is a model of $C \sqsubseteq D$, in symbols $\mathcal{T} \models C \sqsubseteq D$. We use $\models C \sqsubseteq D$ to denote that $C \sqsubseteq D$ follows from the empty TBox. Finally, a TBox $\mathcal{T}'$ *follows from* (or is *entailed by*) a TBox $\mathcal{T}$ if every model of $\mathcal{T}$ is a model of $\mathcal{T}'$, in symbols $\mathcal{T} \models \mathcal{T}'$.

We now introduce the main notion that we study in this paper.

Definition 1. *Let $\mathcal{T}$ be an $\mathcal{ALC}$-TBox and let $\Sigma \subseteq \mathsf{sig}(\mathcal{T})$ be a signature. We say that an $\mathcal{ALC}$-TBox $\mathcal{T}_\Sigma$ is a* Σ-uniform interpolant *of the TBox $\mathcal{T}$ iff $\mathsf{sig}(\mathcal{T}_\Sigma) \subseteq \Sigma$, $\mathcal{T} \models \mathcal{T}_\Sigma$, and for every $\mathcal{ALC}$ Σ-concept inclusion $C \sqsubseteq D$ with $\mathcal{T} \models C \sqsubseteq D$ it holds that $\mathcal{T}_\Sigma \models C \sqsubseteq D$.*

Uniform interpolation can be seen as the dual notion of *forgetting*: a TBox $\mathcal{T}_\Upsilon$ is the result of forgetting about a signature Υ in a TBox $\mathcal{T}$ iff $\mathcal{T}_\Upsilon$ is a uniform interpolant of $\mathcal{T}$ w.r.t. $\Sigma = \mathsf{sig}(\mathcal{T}) \setminus \Upsilon$. As the following example shows, uniform interpolants of $\mathcal{ALC}$-TBoxes do not always exist.

Example 2. *Let $\mathcal{T} = \{A \sqsubseteq B,\ B \sqsubseteq C \sqcap \exists r.B\}$ and $\Sigma = \{A, C, r\}$. Then there does not exist a Σ-uniform interpolant of $\mathcal{T}$ as (in particular) the infinite number of consequences of the form $A \sqsubseteq \exists r.C$, $A \sqsubseteq \exists r.\exists r.C, \dots$ cannot be captured by an $\mathcal{ALC}$-TBox $\mathcal{T}'$ with $\mathsf{sig}(\mathcal{T}') \subseteq \Sigma$. On the other hand, for $\mathcal{T}' = \{A \sqsubseteq B,\ B \sqsubseteq C \sqcap \exists r.B,\ D \equiv B\}$ and $\Sigma' = \{A, C, D, r\}$, a Σ'-uniform interpolant of $\mathcal{T}'$ is $\{A \sqsubseteq D,\ D \sqsubseteq C \sqcap \exists r.D\}$.*

Uniform interpolation is also related to the notion of logical difference between ontologies.

Definition 3. *The Σ-logical difference between $\mathcal{ALC}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ is the set $\mathsf{Diff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ of all $\mathcal{ALC}$-concept inclusions $C \sqsubseteq D$ such that $\mathsf{sig}(C \sqsubseteq D) \subseteq \Sigma$, $\mathcal{T}_1 \models C \sqsubseteq D$, and $\mathcal{T}_2 \not\models C \sqsubseteq D$.*

It is easy to see that $\mathsf{Diff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$ if, and only if, $\mathcal{T}_2 \models \mathcal{T}_1^{(\Sigma)}$ where $\mathcal{T}_1^{(\Sigma)}$ is a Σ-uniform interpolant of $\mathcal{T}_1$. Moreover, if $\mathcal{T}_2 \not\models \mathcal{T}_1^{(\Sigma)}$, every inclusion $C \sqsubseteq D \in \mathcal{T}_1^{(\Sigma)}$ with $\mathcal{T}_2 \not\models C \sqsubseteq D$ can be regarded as a *witness* of $\mathsf{Diff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$.

With the exception of acyclic $\mathcal{EL}$-TBoxes, checking whether the logical difference between two ontologies is nonempty is at least one exponential harder than reasoning (Konev et al. 2012). Additionally, if the set $\mathsf{Diff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ is nonempty, it is typically infinite. Therefore, in practice, the notion of logical difference is primarily used as a theoretical underpinning of its *approximations* that limit the choice of inclusions $C \sqsubseteq D$ in Definition 3 to Σ-inclusions constructed according to some syntactic rules, see e.g. (Jiménez-Ruiz et al. 2009), (Gonçalves, Parsia, and Sattler 2012a; 2012b).

Computing Uniform Interpolants by $\mathcal{ALC}$-Resolution

The aim of our work is to investigate a practical approach for computing uniform interpolants when they exist. Note that the procedure given in (Lutz and Wolter 2011) is inherently inefficient as it requires one to explicitly construct the double-exponential size internalisation $C_\mathcal{T}$ of a given TBox $\mathcal{T}$.

Our approach is to introduce a resolution-like calculus for $\mathcal{ALC}$ that derives consequences of a TBox $\mathcal{T}$ such that a concept inclusion $C \sqsubseteq D$ is entailed by $\mathcal{T}$ iff a contradiction can be derived from $\mathcal{T}$ and $C \sqcap \neg D$. Similarly to (Herzig and Mengin 2008), we then show that any derivation can be restructured in such a way that inferences on selected concept names always precede inferences on other concept names. Then, if the signature Σ is such that $\mathsf{sig}(\mathcal{T}) \setminus \Sigma$ only contains concept names, we generate a set of Σ-consequences $\mathcal{T}'$ of $\mathcal{T}$ by applying the inference rules in a forward chaining manner such that for an arbitrary Σ-inclusion $C \sqsubseteq D$ a contradiction can be derived from $\mathcal{T}$ and $C \sqcap \neg D$ iff a contradiction can be derived from $\mathcal{T}'$ and $C \sqcap \neg D$. Thus, if the forward-chaining process terminates, $\mathcal{T}'$ is a Σ-uniform-interpolant for $\mathcal{T}$.

$\mathcal{ALC}$-Resolution. $\mathcal{ALC}$-resolution operates on $\mathcal{ALC}$ formulae in conjunctive normal form defined according to the following grammar (this is similar to (Herzig and Mengin 2008)):

$$
\begin{array}{rcl}
\text{Literal} & ::= & A \mid \neg A \mid \forall r.\text{Clause} \mid \exists r.\text{CNF} \\
\text{Clause} & ::= & \text{Literal} \mid \text{Clause} \sqcup \text{Clause} \mid \bot \\
\text{CNF} & ::= & \top \mid \text{Clause} \mid \text{Clause} \sqcap \text{CNF}
\end{array}
$$

To simplify the presentation, we assume that clauses are sets of literals and that CNF expressions are sets of clauses. Then $\bot$ corresponds to the empty clause and $\top$ to the empty set of clauses. In the following, the calligraphic letters $\mathcal{C}, \mathcal{D}, \mathcal{E}$ symbolise clauses and $\mathcal{F}, \mathcal{G}$ represent sets of clauses. Similarly to first-order formulae, every $\mathcal{ALC}$ concept can be transformed into an equivalent set of $\mathcal{ALC}$ clauses. The depth of a clause $\mathcal{C}$, $\mathrm{Depth}(\mathcal{C})$, is defined to be the maximal nesting depth of the quantifiers contained in $\mathcal{C}$.

We additionally assume that every clause is assigned a *type*. Clauses obtained from the clausification of TBox inclusions are of the type *universal*, and clauses resulting from the clausification of inclusions to be tested for entailment by the TBox are of the type *initial*. The type of a derived clause is determined by the types of the clauses from which it is derived and by the derivation rule that is used.

Example 4. *The clausification of $\mathcal{T}$ from Example 2 produces three universal clauses: $\neg A \sqcup B$, $\neg B \sqcup C$, $\neg B \sqcup \exists r.B$.*

We now introduce the two resolution calculi $\mathfrak{T}$ and $\mathfrak{T}^u$. The former calculus assumes the TBox to be empty, whereas the latter takes TBox inclusions into account. Thus, $\mathfrak{T}$ derives the empty clause from the set of initial clauses stemming from the clausification of an inclusion $\top \sqsubseteq C \sqcap \neg D$ iff $\models C \sqsubseteq D$; and $\mathfrak{T}^u$ derives the empty clause from the universal clauses stemming from the clausification of a TBox $\mathcal{T}$ and the initial clauses stemming from the clausification of an inclusion $\top \sqsubseteq C \sqcap \neg D$ iff $\mathcal{T} \models C \sqsubseteq D$.

The calculus $\mathfrak{T}$ is defined with the help of the relation $\Rightarrow_\alpha$ given in Fig. 1. For every $\alpha \in \mathsf{N_C} \cup \{\bot\}$, the relation $\Rightarrow_\alpha$ associates with a set of clauses $\mathcal{N}$ a new clause $\mathcal{C}$ which can be 'derived' from the set $\mathcal{N}$ by 'resolving' on α. $\mathfrak{T}$ now consists of the following two inference rules.

$$
\frac{\mathcal{C}}{\mathcal{E}}\ (\text{if } \mathcal{C} \Rightarrow_\alpha \mathcal{E}) \qquad\qquad \frac{\mathcal{C} \quad \mathcal{D}}{\mathcal{E}}\ (\text{if } \mathcal{C}, \mathcal{D} \Rightarrow_\alpha \mathcal{E}),
$$

where $\mathcal{C}, \mathcal{D}$, and $\mathcal{E}$ are initial clauses.

The calculus $\mathfrak{T}^u$ operates initial and universal clauses and also consists of two rules:

$$
\frac{\mathcal{C}}{\mathcal{E}}\ (\text{if } \mathcal{C} \Rightarrow_\alpha \mathcal{E}) \qquad\qquad \frac{\mathcal{C}' \quad \mathcal{D}}{\mathcal{E}'}\ (\text{if } \mathcal{C}', \mathcal{D} \Rightarrow_\alpha^u \mathcal{E}'),
$$

where $\mathcal{C}, \mathcal{C}', \mathcal{D}$ are initial or universal clauses, and $\mathcal{C}', \mathcal{D} \Rightarrow_\alpha^u \mathcal{E}'$ holds iff either $\mathcal{C}', \mathcal{D} \Rightarrow_\alpha \mathcal{E}'$, or $\mathcal{D}$ is a universal clause and there exist role names $r_1, \ldots, r_n \in \mathsf{N_R}$ $(n \geq 1)$ such that $\mathcal{C}', \forall r_1. \ldots \forall r_n.\mathcal{D} \Rightarrow_\alpha \mathcal{E}'$. (Intuitively, the calculus $\mathfrak{T}^u$ allows for inferences with universal clauses at arbitrary nesting levels of quantifiers, which the calculus $\mathfrak{T}$ does not.) Then $\mathcal{E}$ is a universal clause if $\mathcal{C}$ is a universal clause, and an initial clause otherwise. Similarly, $\mathcal{E}'$ is a universal clause if both $\mathcal{C}'$ and $\mathcal{D}$ are universal clauses, and an initial clause otherwise.

We assume that every clause $\mathcal{E}$ that results from a $\mathfrak{T}$- or $\mathfrak{T}^u$-inference is implicitly simplified by exhaustively removing all occurrences of literals of the form $\exists r.(\mathcal{F}, \bot)$.

Example 5. *For the universal clauses from Example 4, we have for instance,*

$$
\neg A \sqcup B, \neg B \sqcup \exists r.B \ \Rightarrow_B \ \neg A \sqcup \exists r.B \quad \textit{by (rule A)}.
$$

So, the universal clause $\neg A \sqcup \exists r.B$ is derivable by $\mathfrak{T}^u$ from $\neg A \sqcup B$ and $\neg B \sqcup \exists r.B$. As $\neg B \sqcup C$ is a universal clause and

$$
\neg B \sqcup \exists r.B, \forall r.\neg B \sqcup C \Rightarrow_B \neg B \sqcup \exists r.(B, C) \quad \textit{by (rule } \forall\exists\textit{)},
$$

(rule $\bot$)	$\mathcal{C}'_1 \sqcup \forall r.\bot,\ \ \mathcal{C}'_2 \sqcup \exists r.\mathcal{F} \Longrightarrow_\bot \mathcal{C}'_1 \sqcup \mathcal{C}'_2$	
(rule A)	$\mathcal{C}'_1 \sqcup A,\ \ \mathcal{C}'_2 \sqcup \neg A \Longrightarrow_A \mathcal{C}'_1 \sqcup \mathcal{C}'_2$	
(rule $\forall\exists$)	$\mathcal{C}'_1 \sqcup \forall r.\mathcal{C}_1,\ \ \mathcal{C}'_2 \sqcup \exists r.(\mathcal{C}_2, \mathcal{F}) \Longrightarrow_\alpha \mathcal{C}'_1 \sqcup \mathcal{C}'_2 \sqcup \exists r.(\mathcal{C}_2, \mathcal{F}, \mathcal{C}_3)$,	if $\mathcal{C}_1, \mathcal{C}_2 \Longrightarrow_\alpha \mathcal{C}_3$
(rule $\forall\forall$)	$\mathcal{C}'_1 \sqcup \forall r.\mathcal{C}_1,\ \ \mathcal{C}'_2 \sqcup \forall r.\mathcal{C}_2 \Longrightarrow_\alpha \mathcal{C}'_1 \sqcup \mathcal{C}'_2 \sqcup \forall r.\mathcal{C}_3$,	if $\mathcal{C}_1, \mathcal{C}_2 \Longrightarrow_\alpha \mathcal{C}_3$
(rule $\exists_1$)	$\mathcal{C}' \sqcup \exists r.(\mathcal{C}_1, \mathcal{F}) \Longrightarrow_\alpha \mathcal{C}' \sqcup \exists r.(\mathcal{C}_1, \mathcal{F}, \mathcal{C}_2)$,	if $\mathcal{C}_1 \Longrightarrow_\alpha \mathcal{C}_2$
(rule $\exists_2$)	$\mathcal{C}' \sqcup \exists r.(\mathcal{C}_1, \mathcal{C}_2, \mathcal{F}) \Longrightarrow_\alpha \mathcal{C}' \sqcup \exists r.(\mathcal{C}_1, \mathcal{C}_2, \mathcal{F}, \mathcal{C}_3)$,	if $\mathcal{C}_1, \mathcal{C}_2 \Longrightarrow_\alpha \mathcal{C}_3$
(rule $\forall$)	$\mathcal{C}' \sqcup \forall r.\mathcal{C}_1 \Longrightarrow_\alpha \mathcal{C}' \sqcup \forall r.\mathcal{C}_2$,	if $\mathcal{C}_1 \Longrightarrow_\alpha \mathcal{C}_2$

Figure 1: Rules of $\Longrightarrow_\alpha$.

the universal clause $\neg B \sqcup \exists r.(B, C)$ is derivable by $\mathfrak{T}^u$ from $\neg B \sqcup \exists r.B$ and $\neg B \sqcup C$. By applying the inference rules to old and newly generated clauses, one can conclude that the universal clauses $\neg A \sqcup \exists r.(B, C)$ and $\neg A \sqcup \exists r.(B, \exists r.B)$ are also derivable by $\mathfrak{T}^u$ from $\mathcal{N} = \{\neg A \sqcup B,\ \neg B \sqcup C,\ \neg B \sqcup \exists r.B\}$.

For $x \in \{\mathfrak{T}, \mathfrak{T}^u\}$, a x-derivation (tree) Δ built from a set of clauses $\mathcal{N}$ is a finite binary tree where each leaf is labelled with a clause from $\mathcal{N}$ and each non-leaf node n is labelled with a clause $\mathcal{C}$ such that $\mathcal{C}$ results from an x-inference on the parent(s) of n in Δ. We say that Δ is a derivation of a clause $\mathcal{C}$ if the root of Δ is labelled with $\mathcal{C}$. A derivation of the empty clause is called a *refutation*. Every path $n_1, \ldots, n_m$ of nodes in Δ where n_1 is a leaf node and n_m is the root node induces an *inference path* $\alpha_2, \ldots, \alpha_m$, where $\alpha_i \in \mathsf{N_C} \cup \{\bot\}$ ($2 \le i \le m$) denotes the concept name, or $\bot$, which has been resolved upon to obtain the clause that is the label of the node n_i. For a signature $\Upsilon \subseteq \mathsf{N_C}$ and a strict total order $\succ \subseteq \Upsilon \times \Upsilon$, a derivation Δ is a $(x, \Upsilon, \succ)$-derivation if for every inference path $\alpha_1, \ldots, \alpha_n$ of Δ (with $\alpha_i \in \mathsf{N_C} \cup \{\bot\}$ for every $1 \le i \le n$) there exists $0 \le k \le n$ such that $\{\alpha_1, \ldots, \alpha_k\} \subseteq \Upsilon$, $\alpha_j \succ \alpha_{j+1}$ or $\alpha_j = \alpha_{j+1}$ for every $1 \le j < k$, and $\alpha_j \notin \Upsilon$ for every $k < j \le n$.

We prove that for every unsatisfiable set of initial clauses there always exists a $(\mathfrak{T}, \Upsilon, \succ)$-refutation by extending the results and proof methods of (Herzig and Mengin 2008).

Theorem 6 ($\mathfrak{T}$-Completeness)**.** *Let $\Upsilon \subseteq \mathsf{N_C}$, let $\succ \subseteq \Upsilon \times \Upsilon$ be a strict total order on Υ and let C and D be $\mathcal{ALC}$ concepts. Then it holds that $\models C \sqsubseteq D$ iff there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation of the empty clause from the initial clauses $\mathrm{Cls}(C \sqcap \neg D)$.*

A weaker version of this result, stating that any derivation in $\mathfrak{T}$ can be reordered so that inferences on concept names from Υ always precede inferences on other concept names, or $\bot$, has been previously announced in (Herzig and Mengin 2008); however, as we show in the full version of the paper, the proof appears to have some gaps.

To prove completeness for $\mathfrak{T}^u$, we observe the following link between derivations in $\mathfrak{T}$ and $\mathfrak{T}^u$. Let $\mathcal{N}$ be a set of clauses and let

$$\begin{aligned}&\mathrm{Univ}_0(\mathcal{N}) = \mathcal{N};\\ &\mathrm{Univ}_{i+1}(\mathcal{N}) = \mathrm{Univ}_i(\mathcal{N}) \cup \\ &\qquad\qquad \textstyle\bigcup_{r \in \mathsf{N_R} \cap \mathsf{sig}(\mathcal{N})} \{\, \forall r.\mathcal{C} \mid \mathcal{C} \in \mathrm{Univ}_i(\mathcal{N}) \,\}\end{aligned}$$

and $\mathrm{Univ}(\mathcal{N}) = \bigcup_{i \ge 0} \mathrm{Univ}_i(\mathcal{N})$.

Theorem 7. *Let $\mathcal{M}$ be a set of initial clauses and let $\mathcal{N}$ be a set of universal clauses. Additionally, let Δ be a $(\mathfrak{T}, \Upsilon, \succ)$-refutation from $\mathcal{M} \cup \mathrm{Univ}(\mathcal{N})$ such that there exists $n \in \mathbb{N}$ with $\mathrm{Depth}(\mathcal{C}) \le n$ for every $\mathcal{C} \in$ Clauses(Δ). Then there exists a $(\mathfrak{T}^u, \Upsilon, \succ)$-derivation Δ^u of the empty clause from $\mathcal{M} \cup \mathcal{N}$ such that $\mathrm{Depth}(\mathcal{C}) \le n$ for every $\mathcal{C} \in$ Clauses(Δ^u).*

We then use Theorems 6 and 7 and the fact that every $\mathcal{ALC}$-TBox can be internalised. Notice that the actual TBox internalisation $C_\mathcal{T}$ does not have to be computed as it is only used for the proof of completeness.

Corollary 8 ($\mathfrak{T}^u$-Completeness)**.** *Let $\mathcal{T}$ be an $\mathcal{ALC}$-TBox, let $\Upsilon \subseteq \mathsf{N_C}$, let $\succ \subseteq \Upsilon \times \Upsilon$ be a strict total order on Υ and let C and D be $\mathcal{ALC}$ concepts. Then it holds that $\mathcal{T} \models C \sqsubseteq D$ iff there exists a $(\mathfrak{T}^u, \Upsilon, \succ)$-derivation of the empty clause from the universal clauses $\mathrm{Cls}(\mathcal{T})$ and the initial clauses $\mathrm{Cls}(C \sqcap \neg D)$.*

Computing Uniform Interpolants. The procedure UNIFORMINTERPOLANT depicted in Algorithm 1 takes as input an $\mathcal{ALC}$-TBox $\mathcal{T}$, a signature $\Sigma \subseteq \mathsf{sig}(\mathcal{T})$ such that $\Sigma \cap \mathsf{N_R} = \mathsf{sig}(\mathcal{T}) \cap \mathsf{N_R}$ and a strict total order $\succ \subseteq \Upsilon \times \Upsilon$ over $\Upsilon = \mathsf{sig}(\mathcal{T}) \setminus \Sigma$. Following the outline of (Herzig and Mengin 2008), after the clausification of $\mathcal{T}$, the procedure iterates over the concept names contained in Υ in descending order according to the relation $\succ$. In each iteration the clause set $\mathcal{N}$ is expanded with all possible $\mathfrak{T}^u$-inferences on the current concept name $A \in \Upsilon$. Finally, after iterating over all the concept names from $\Upsilon = \mathsf{sig}(\mathcal{T}) \setminus \Sigma$, the operator 'Supp' is applied on the resulting clauses, which replaces all occurrences of Υ concept names in clauses with $\top$ and then simplifies the resulting CNF.

Example 9. *For the clauses obtained in Example 5,* $\mathrm{Supp}(\{B\}, \neg A \sqcup C) = \neg A \sqcup C$, $\mathrm{Supp}(\{B\}, \neg A \sqcup \exists r.B) = \neg A \sqcup \exists r.\top$, $\mathrm{Supp}(\{B\}, \neg A \sqcup \exists r.(B, C)) = \neg A \sqcup \exists r.C$.

One can show that if Algorithm 1 terminates, for all $\mathcal{ALC}$ Σ-concepts C, D such that there exists a $(\mathfrak{T}^u, \Upsilon, \succ)$-refutation Δ^u from the universal clauses $\mathrm{Cls}(\mathcal{T})$ and the initial clauses $\mathrm{Cls}(C \sqcap \neg D)$ it holds that $\mathcal{F}_\Sigma(\mathcal{T}) \models C \sqsubseteq D$. Thus, it follows from Corollary 8 that if Algorithm 1 terminates, it computes a Σ-uniform interpolant of $\mathcal{T}$. However, Algorithm 1 does not terminate if a uniform interpolant does not exist. For example, when applied to $\mathcal{T}$ from Example 2, Algorithm 1 can generate, among others, the infinite sequence of universal clauses $\neg A \sqcup \exists r.C$, $\neg A \sqcup \exists r.(C, \exists r.C), \ldots$ and so it does not terminate. Moreover,

Algorithm 1

```
1:  procedure UniformInterpolant(𝒯, Σ, ≻)
2:      ϒ := sig(𝒯) \ Σ
3:      𝒩 := Cls(𝒯)
4:      while ϒ ≠ ∅ do
5:          A := max_≻(ϒ)
6:          𝒩 := Res^∞_{𝔗^u,{A}}(𝒩)
7:          ϒ := ϒ \ {A}
8:      end while
9:      return ℱ_Σ(𝒯) = Supp(sig(𝒯) \ Σ, 𝒩)
10: end procedure
```

as the TBox $\mathcal{T}$ from Example 2 is a subset of $\mathcal{T}'$, and so $\mathrm{Cls}(\mathcal{T}) \subseteq \mathrm{Cls}(\mathcal{T}')$, Algorithm 1 will derive, among others, the same clauses when it is applied on $\mathcal{T}'$. Thus, in some cases Algorithm 1 does not terminate even though a uniform interpolant exists.

To guarantee termination on all inputs, we focus on the notion of depth-bounded uniform interpolation (related to the notion of 'bounded forgetting' (Zhou and Zhang 2011)). Let $\mathcal{T}$ be an $\mathcal{ALC}$-TBox and let $\Sigma \subseteq \mathsf{sig}(\mathcal{T})$ be a signature. We say that an $\mathcal{ALC}$-TBox $\mathcal{T}_\Sigma$ is a *depth n-bounded uniform interpolant* of the TBox $\mathcal{T}$ w.r.t. Σ iff $\mathsf{sig}(\mathcal{T}_\Sigma) \subseteq \Sigma$, $\mathcal{T} \models \mathcal{T}_\Sigma$, and for every $\mathcal{ALC}$ Σ-concept inclusion $C \sqsubseteq D$ with $\mathcal{T} \models C \sqsubseteq D$ and $\max\{\mathrm{Depth}(C), \mathrm{Depth}(D)\} \leq n$ it holds that $\mathcal{T}_\Sigma \models C \sqsubseteq D$. Let $\mathcal{F}_{\Sigma,m}(\mathcal{T})$ be the outcome of Algorithm 1 where in Step 6 only clauses up to depth m are generated. The following example shows that it might be necessary to consider intermediate clauses of a depth $m > n$ in order to preserve all the Σ-consequences of depth n entailed by $\mathcal{T}$.

Example 10. *Let* $\mathcal{T} = \{A \sqsubseteq \exists r.C,\ C \sqsubseteq \exists s.\top,\ \neg B \sqsubseteq \forall s.\bot\}$, $\Sigma = \{A, B, r, s\}$, $\Upsilon = \{C\}$ *and* $\succ = \emptyset$. *Then every* $(\mathfrak{T}^u, \Upsilon, \succ)$*-refutation from the universal clauses* $\mathrm{Cls}(\mathcal{T})$ *and the initial clauses* $\{A, \forall r.\neg B\}$ *derives the clause* $\neg A \sqcup \exists r.(C, \exists s.\top)$.

We establish, however, that by choosing the maximal depth of derived clauses appropriately, the procedure depicted in Algorithm 1 computes uniform interpolants that preserve consequences up to a specified depth n.

Theorem 11. *Let $\mathcal{T}$ be an $\mathcal{ALC}$-TBox,* $\Sigma \subseteq \mathsf{sig}(\mathcal{T})$ *a signature such that* $\Sigma \cap \mathsf{N_R} = \mathsf{sig}(\mathcal{T}) \cap \mathsf{N_R}$, *and let* $n \geq 0$. *Set* $m = n + 2^{|\mathsf{sub}(\mathrm{Cls}(\mathcal{T}))|+1} + \max\{\, \mathrm{Depth}(\mathcal{C}) \mid \mathcal{C} \in \mathrm{Cls}(\mathcal{T}) \,\}$, *where* $\mathsf{sub}(\mathrm{Cls}(\mathcal{T}))$ *is the set of subconcepts of* $\mathrm{Cls}(\mathcal{T})$. *Then it holds that* $\mathcal{F}_{\Sigma,m}(\mathcal{T})$ *is a depth n-bounded uniform interpolant of the TBox $\mathcal{T}$ w.r.t. Σ.*

We can combine this result with the results of (Lutz and Wolter 2011): for any $\mathcal{ALC}$-TBox $\mathcal{T}$ and signature Σ, if a Σ-uniform interpolant of $\mathcal{T}$ exists, then there exists a uniform interpolant of depth bounded by $2^{2^{|\mathcal{T}|+1}} + 1$. Thus, if $\Sigma \cap \mathsf{N_R} = \mathsf{sig}(\mathcal{T}) \cap \mathsf{N_R}$, there exists m, which can be computed based on the bound in Theorem 11 and the results of (Lutz and Wolter 2011), such that $\mathcal{F}_{\Sigma,m}(\mathcal{T})$ is a Σ-uniform interpolant of $\mathcal{T}$.

The bound in Theorem 11 can be significantly improved if the TBox is acyclic. For an acyclic $\mathcal{ALC}$-TBox $\mathcal{T}$ we define $\mathrm{ExpansionDepth}(\mathcal{T}) = \max\{\, \mathrm{Depth}(A[\mathcal{T}]) \mid A \in \mathsf{sig}(\mathcal{T}) \,\}$, where $A[\mathcal{T}]$ denotes the concept obtained by exhaustively replacing every concept B with C_B if $B \sqsubseteq C_B \in \mathcal{T}$ or $B \equiv C_B \in \mathcal{T}$.

Theorem 12. *Let $\mathcal{T}$ be an acyclic $\mathcal{ALC}$ TBox,* $\Sigma \subseteq \mathsf{sig}(\mathcal{T})$ *a signature such that* $\Sigma \cap \mathsf{N_R} = \mathsf{sig}(\mathcal{T}) \cap \mathsf{N_R}$, *and let* $n \geq 0$. *Set* $m = \mathrm{ExpansionDepth}(\mathcal{T}) + n$. *Then it holds that* $\mathcal{F}_{\Sigma,m}(\mathcal{T})$ *is a uniform interpolant limited to consequence depth n of the TBox $\mathcal{T}$ w.r.t. Σ.*

Note that in the description logic $\mathcal{EL}$ (i.e. the fragment of $\mathcal{ALC}$ that does not allow $\bot$, negation, disjunction, or universal quantification) the acyclicity of a TBox guarantees the existence of uniform interpolants (Konev, Walther, and Wolter 2009) for any signature Σ. Interestingly, this is not true in the case of $\mathcal{ALC}$. Moreover, as the following example shows, there exists an acyclic $\mathcal{EL}$-TBox $\mathcal{T}$ and a signature Σ for which no $\mathcal{ALC}$ Σ-uniform interpolant exists.

Example 13. *Consider* $\Sigma = \{A, A_0, A_1, A_2, E, r\}$ *and* $\mathcal{T} = \{A \sqsubseteq \exists r.B,\ A_0 \sqsubseteq \exists r.(A_1 \sqcap B),\ E \equiv A_1 \sqcap B \sqcap \exists r.(A_2 \sqcap B)\}$. *Then for every* $n \geq 0$, $\mathcal{T}$ *entails the inclusion*

$$A_0 \sqcap \prod_{i=1}^{n} \underbrace{\forall r. \ldots \forall r.}_{i} (A \sqcap \neg E \sqcap (A_1 \sqcup A_2)) \sqsubseteq \underbrace{\exists r. \ldots \exists r.}_{n} A_1.$$

This infinite sequence of $\mathcal{ALC}$ consequences of $\mathcal{T}$ cannot be captured by any $\mathcal{ALC}$ Σ-TBox $\mathcal{T}'$, which can be proved formally using Theorem 9 in (Lutz and Wolter 2011).

Case Study

We have implemented a prototype of an inference computation architecture using the calculus $\mathfrak{T}^u$ and the inference relation $\Rightarrow_\alpha$ in Java. It has turned out that our initial implementation of Algorithm 1 did not perform well in practice. This was in particular due to the fact that clauses can contain sets $\mathcal{F}$ of other clauses in existential literals $\exists r.\mathcal{F}$, which renders all the possible inferences on clauses from $\mathcal{F}$ 'explicit'. For example, if we resolve the universal clause which just consists of the existential literal $\exists r.(A)$ with the universal clauses $\neg A \sqcup B_1, \ldots, \neg A \sqcup B_n$ on the concept name A, then not only the clauses $\exists r.(A, B_1)$, $\exists r.(A, B_2)$, $\exists r.(A, B_3)$,... could be derived but all clauses of the form $\exists r.(A, \mathcal{G})$, where $\mathcal{G}$ is a subset of $\{B_1, \ldots, B_n\}$.

A common technique to reduce the number of inferences that have to be made is to use forward- and backward deletion of subsumed clauses (Bachmair and Ganzinger 2001). However, it is known (Auffray, Enjalbert, and Hébrard 1990) that the subsumption lemma (stating that if a clause $\mathcal{E}$ results from an inference involving two clauses $\mathcal{C}$ and $\mathcal{D}$, and if there exist clauses $\mathcal{C}'$, $\mathcal{D}'$ such that $\mathcal{C}'$ subsumes $\mathcal{C}$ and $\mathcal{D}'$ subsumes $\mathcal{D}$, then either $\mathcal{E}$ is subsumed by one of $\mathcal{C}'$, $\mathcal{D}'$, or a clause $\mathcal{E}'$ can be derived from $\mathcal{C}'$ and $\mathcal{D}'$ such that $\mathcal{E}'$ subsumes $\mathcal{E}$) does not hold even in the modal logic K for the standard *minimal subsumption relation* $\leq_s$ (Auffray, Enjalbert, and Hébrard 1990) and $\Rightarrow_\alpha$. To be able to prove that one can safely discard subsumed clauses, we have modified the inference relation $\Rightarrow_\alpha$ by introducing the following additional rule (rule $\exists_f$)

$$\mathcal{C}_1 \sqcup \forall r.\mathcal{D},\ \ \mathcal{C}_2 \sqcup \exists r.\mathcal{F} \Longrightarrow_{\exists_f} \mathcal{C}_1 \sqcup \mathcal{C}_2 \sqcup \exists r.(\mathcal{F}, \mathcal{D}).$$

	Uniform Interpolation				Forgetting					
	$\lvert\Sigma \cap \mathsf{N_C}\rvert = 5$		$\lvert\Sigma \cap \mathsf{N_C}\rvert = 10$		$\lvert\Upsilon\rvert = 10$		$\lvert\Upsilon\rvert = 15$		$\lvert\Upsilon\rvert = 25$	
	Success Rate (%)	Avrg # Axioms	Success Rate (%)	Avrg # Axioms	Success Rate (%)	Avrg # axioms	Success Rate (%)	Avrg # Axioms	Success Rate (%)	Avrg # Axioms
AMINO-ACID v1.2	100	61.40	92	143.35	100	645.67	87	665.24	64	396.98
BHO v0.4	71	30.01	16	52.43	99	2374.73	99	2363.42	91	2383.96
CAO v1.4	100	279.02	100	283.33	100	369.54	100	369.22	10	366.07
CDAO	100	288.21	100	288.42	100	293.48	100	293.41	10	293.02
CHEMBIO v1.1	92	7 1.89	60	94.40	100	295.85	100	293.09	10	293.64
CPRO v0.85	100	585.08	100	533.82	100	307.76	100	309.46	10	316.31
DDI v0.9	100	249.80	100	259.41	100	276.27	100	278.55	10	276.61
DIKB v1.4	2	1591.50	0	-	97	622.67	83	689.44	56	816.39
GRO v0.5	0	-	0	-	94	959.85	91	940.03	79	997.59
IDO	0	-	0	-	94	1202.71	90	1203.78	80	1215.36
LIPRO v1.1	73	7.93	58	13.22	91	2287.24	58	2381.43	45	2297.37
NCI v08.10e	23	887.34	1	1397.00	97	100693.26	98	100611.60	99	100889.50
NEOMARK v4.1	31	19.45	14	27.28	100	338.52	100	333.26	10	324.86
OMRSE	100	485.00	100	485.00	100	485.00	100	485.00	10	485.00
OBIWS v1.1	100	112.56	100	118.70	100	189.66	100	187.71	10	184.13
ONTODM v1.1.1	0	-	0	-	98	1711.40	98	1704.67	93	1693.61
OPL	100	829.41	100	832.93	100	848.60	100	848.99	10	848.73
PROPREO v1.1	41	2.07	19	31.84	100	561.43	100	560.85	99	578.08
RNAO r113	100	355.86	100	362.83	100	439.64	100	439.10	10	439.71
SAO v1.2.4	0	-	0	-	99	2702.23	100	2700.85	98	2715.30
SITBAC v1.3	0	-	0	-	93	508.40	93	537.48	79	595.51
TOK v0.2.1	0	-	0	-	97	496.12	93	529.06	72	567.11
VSO	0	-	0	-	83	348.87	79	397.65	50	371.38

Table 1: Uniform Interpolation and Forgetting for BioPortal Ontologies on Small Signatures.

We will denote the resulting inference relation by $\Rightarrow^f_\alpha$ with $\alpha \in \mathsf{N_C} \cup \{\bot, \exists_f\}$. One can then prove that a variant of the subsumption lemma holds for the relations $\leq_s$ and $\Rightarrow^f_\alpha$, which allows us to employ forward- and backward deletion of subsumed clauses in our implementation.

In order to further speed up computations, we first extract the locality-based $\top\bot^*$ Σ-module (Cuenca Grau et al. 2008; Sattler, Schneider, and Zakharyaschev 2009) for a given TBox $\mathcal{T}$. The locality-based module entails the same Σ-inclusions as the TBox $\mathcal{T}$ but it is often considerably smaller in size. We also rely on ontologies to have structure: if a concept name occurs in several inclusions, it is likely that it occurs in the same syntactic pattern. Thus,

1. if the clause set contains some clauses $\mathcal{C}_1 \sqcup \mathcal{D}_\Upsilon, \ldots, \mathcal{C}_m \sqcup \mathcal{D}_\Upsilon$ such that for every $1 \leq i \leq m$ we have $\mathsf{sig}(\mathcal{C}_i) \cap \Upsilon = \emptyset$, we rewrite them into $X \sqcup \mathcal{D}_\Upsilon$, where $X \equiv \mathcal{C}_1 \sqcap \ldots \sqcap \mathcal{C}_m$, perform forgetting on Υ symbols and then replace X with its definition.
2. If the clause set contains a clause $\mathcal{C} \sqcup \exists r.(\mathcal{F}_\Upsilon, \mathcal{G}_1) \sqcup \ldots \sqcup \exists r.(\mathcal{F}_\Upsilon, \mathcal{G}_m)$ where $\mathsf{sig}(\mathcal{G}_i) \cap \Upsilon = \emptyset$ for every $1 \leq i \leq m$, we rewrite it into $\mathcal{C} \sqcup \exists r.(\mathcal{F}_\Upsilon, Y)$, where $Y \equiv \mathcal{G}_1 \sqcup \ldots \sqcup \mathcal{G}_m$, perform forgetting on Υ and then replace Y with its definition.

	$\lvert\Sigma \cap \mathsf{N_C}\rvert$	Succes rate (%)	Avrg # axioms
DIKB	5	85	7.482
	10	60	14.033
	15	44	25.114
NCI	5	82	1.62
	10	64	2.65
	50	65	21.369
	100	56	41.089
	150	41	63.146

Table 2: Computing Uniform Interpolants of DIKB v1.4 and of NCI v08.10e Limited to Expansion Depth 3.

Experimental setting. All experiments were conducted on PCs equipped with an Intel Core i5-2500K CPU running at 3.30GHz. 15 GiB of RAM were allocated to the Java VM and an execution timeout of 60 CPU minutes was imposed on each problem. Whenever necessary we pre-processed the ontologies we used for our experiments as follows. For a given ontology $\mathcal{T}$, we first rewrote concept disjointness statements and role domain/range restrictions into $\mathcal{ALC}$ in-

	Computing $\mathsf{Diff}_\Sigma(\mathcal{T}_{i+1}, \mathcal{T}_i)$			Computing $\mathsf{Diff}_\Sigma(\mathcal{T}_i, \mathcal{T}_{i+1})$		
	Successful/ Total Runs	Success Rate (%)	Average # of Witnesses	Successful/ Total Runs	Success Rate (%)	Average # of Witnesses
BDO	3/5	60	12.33	5/5	100	211.40
CHEMINF	25/26	96	7.00	26/26	100	2.26
COGAT	4/4	100	272.00	3/4	75	4.00
JERM	8/13	61	7.00	9/13	69	9.33
NCI	101/108	93	787.10	105/108	97	906.20
NEMO	14/15	93	13.35	15/15	100	33.46
NPO	12/18	66	27.08	12/18	66	5.58
OMRSE	11/11	100	0.54	11/11	100	0.00
OPL	4/4	100	18.75	4/4	100	2.25
SIO	18/35	51	0.00	19/35	54	0.00

Table 3: Computing the Logical Difference between Ontology Versions on their Common Signature.

clusions and then removed any remaining axiom which contained non-$\mathcal{ALC}$ concept (or role) constructors to obtain the $\mathcal{ALC}$-fragment of $\mathcal{T}$. We used Algorithm 1 to forget concept names one by one i.e. for $\mathsf{sig}(\mathcal{T}) \setminus \Sigma = \{A_1, \ldots, A_n\}$, Algorithm 1 was applied iteratively on $A_1, \ldots, A_n$, and we did *not* impose a bound on the depth of clauses; so the computed clause sets contain depth n-bounded uniform interpolants for every $n > 0$. Thus, in all the experiments reported on in this section we computed *true Σ-uniform interpolants* (i.e. not a depth-bounded variant). The correctness of our extensions to Algorithm 1 can be shown by model-theoretic arguments.

Experiments with small signatures. We applied our uniform interpolation tool to compute uniform interpolants w.r.t. small concept signatures $\Sigma \subseteq \mathsf{sig}(\mathcal{T})$ with $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_R} = \Sigma \cap \mathsf{N_R}$ for 21 small to medium size ontologies taken from the BioPortal repository[1]. The number of axioms in the selected ontologies ranges from 192 (for the *Ontology of Medically Related Social Entries*) to 2702 (for the *Subcellular Anatomy Ontology*). To make the experiments more interesting, we also included version 08.10e of the *National Cancer Institute Thesaurus* (NCI). For each considered sample size x and terminology $\mathcal{T}$ we generated 100 signatures Σ by randomly choosing x concept names from $\mathsf{sig}(\mathcal{T})$ and by adding all the role names from $\mathsf{sig}(\mathcal{T})$ to Σ. The results that we obtained are shown in Table 1.

In the left half of Table 1 one can see that the number of successful computations decreased with increasing size of $\Sigma \cap \mathsf{N_C}$, which seems to be due to the fact that the $\top\bot^*$ Σ-modules then contain more symbols that lead to a large number of inferences. Most uniform interpolants that we obtained are relatively small and contain a lot of expressions of the form $\exists r_1 \ldots \exists r_n.\top$. In some cases the process of forgetting certain intermediate concept names generated a few hundred clauses that were simplified or deleted in the remaining computation steps. The success rate, however, varied significantly from one ontology to another. To further investigate this phenomenon, we computed uniform interpolants for a *fragment* of version 08.10e of NCI and for a fragment of version 1.4 of the *Drug Interaction Knowledge Base* (DIKB) that are of expansion depth 3 (that is, we removed all the axioms from both ontologies that led to an expansion depth greater than 3). The resulting DIKB fragment is a small acyclic terminology that contains 120 concept names, 27 roles names, and 127 axioms. The NCI fragment is also an acyclic terminology with 53571 concept names, 78 role names and 62494 axioms (of which 2362 are of the form $A \equiv C$). The results obtained are shown in Table 2. Limiting the expansion depth drastically improved the performance of our prototype implementation with the success rate for signatures containing 5 randomly selected concept names rising from 2% to 85% in the case of DIKB and from 23% to 82% in the case of NCI. For NCI our tool is capable of handling signatures containing up to 150 randomly selected concept names.

As proof of concept for ontology obfuscation, we applied our uniform interpolation tool on (a fragment of) the *Lipid Ontology* (LIPRO) to forget 45 concept names which are intermediate concept names in the ontology's induced concept hierarchy, i.e. those concept names group certain subconcepts together to give structure to the ontology. LIPRO is an acyclic terminology with 593 axioms, 574 concept names and one role name. The maximal size of an axiom is 50. It then took 192 CPU seconds to compute the uniform interpolant, which contains 3415 axioms that have a maximal size of 283. The uniform interpolant that we computed thus approximately contains 6 times more axioms than the original ontology and the maximal axiom size has increased by a factor of 6 as well. Notice that most of the original structure of the ontology has been destroyed while preserving all the consequences entailed by the retained concept names.

Finally, in the right half of Table 1 we report on our success rate for forgetting a small number of concept names. Notice that our prototype implementation performs significantly better in this scenario. This observation suggests that our tool is suitable for checking whether a change made to

[1] All ontologies used for the experiments reported on in this section can be accessed from the BioPortal repository, `http://bioportal.bioontology.org/ontologies`

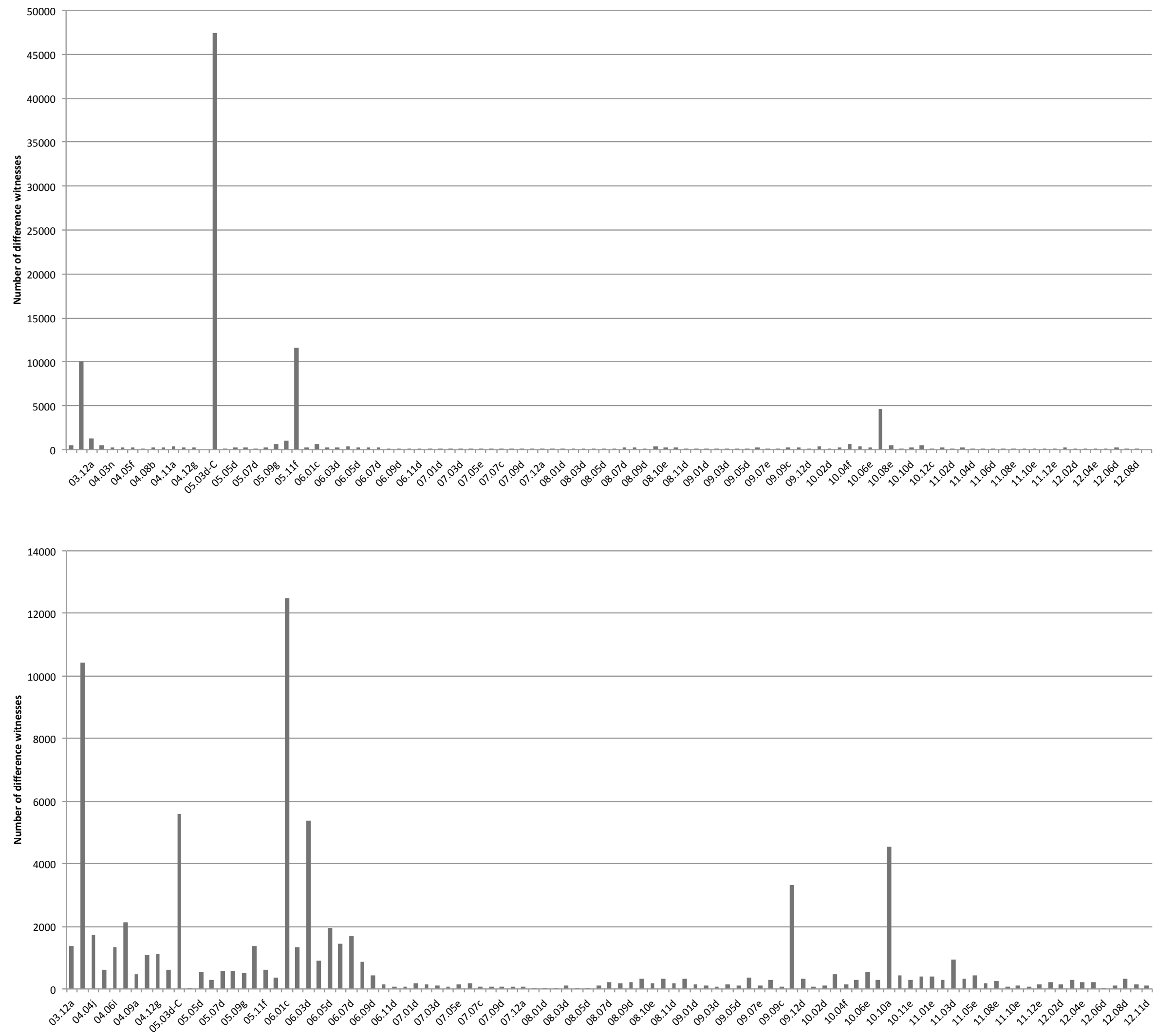


Figure 2: Logical Difference between NCI Versions i and $i+1$ (Top) and Versions $i+1$ and i (Bottom).

an ontology interferes with the meaning of the terms outside the (typically small) fragment under consideration in the context of computing the logical difference between two versions of an ontology.

Applications to Computing the Logical Difference. We selected 10 ontologies that have at least 5 submissions and whose expressivity is at least $\mathcal{ALC}$, including 109 versions of the NCI Thesaurus, from the BioPortal repository.

For every pair of consecutive versions $\mathcal{T}_i$ and $\mathcal{T}_{i+1}$, where version $i+1$ represents the more recent version, and every considered signature Σ, we computed both $\mathsf{Diff}_\Sigma(\mathcal{T}_i, \mathcal{T}_{i+1})$ and $\mathsf{Diff}_\Sigma(\mathcal{T}_{i+1}, \mathcal{T}_i)$. We used the reasoner FaCT++ v1.6.2 (Tsarkov and Horrocks 2006) to determine whether any axiom $C \sqsubseteq D \in \mathcal{T}_i^{(\Sigma)}$ is a witness of $\mathsf{Diff}_\Sigma(\mathcal{T}_i, \mathcal{T}_{i+1})$, where $\mathcal{T}_i^{(\Sigma)}$ is the Σ-uniform interpolant of $\mathcal{T}_i$ computed with our tool (similarly for $\mathsf{Diff}_\Sigma(\mathcal{T}_{i+1}, \mathcal{T}_i)$). Note that the results we obtained are not directly comparable with the logical difference computed for description logics of the $\mathcal{EL}$ family (Konev, Walther, and Wolter 2008; Konev et al. 2012) as illustrated by Example 13.

In our first experiment we used $\Sigma = (\mathsf{sig}(\mathcal{T}_i) \cap \mathsf{sig}(\mathcal{T}_{i+1})) \cup \mathsf{N_R}$. This test captures any change to the meaning of the terms common to both versions. The results of computing the logical difference are given in Table 3. Notice that the success rate of computing $\mathsf{Diff}_\Sigma(\mathcal{T}_i, \mathcal{T}_{i+1})$ was slightly higher than the one of the converse direction. This observation can probably be attributed to the fact that these cases correspond to knowledge contained in an older version being removed from a newer one, which does not seem to happen often.

Interestingly, we could observe one of the highest success rates among all our experiments whilst computing logical differences for distinct versions of NCI. This can possibly be explained by the fact that versions of NCI are released frequently and changes to the ontology are hence introduced

$\lvert(\mathsf{sig}(\mathcal{T}) \setminus \Sigma)$ $\cap \mathsf{N_C}\rvert$	$\mathsf{Diff}_\Sigma(\mathrm{NCI}_{v08.09d}, \mathrm{NCI}_{08.10e})$ Success rate (%)	Avrg # Witnesses	$\mathsf{Diff}_\Sigma(\mathrm{NCI}_{v05.03d}, \mathrm{NCI}_{05.05d})$ Success rate (%)	Avrg # Witnesses	$\mathsf{Diff}_\Sigma(\mathrm{NCI}_{v05.12f}, \mathrm{NCI}_{06.01c})$ Success rate (%)	Avrg # Witnesses
5	100	446.01	100	47 458.14	100	11 564.71
10	99	446.05	100	47 456.66	97	11 595.85
20	100	445.95	100	47 453.26	94	11 671.79
50	88	445.73	100	47 436.72	84	11 849.16
100	88	445.67	100	47 403.76	70	12 468.64
$\lvert(\mathsf{sig}(\mathcal{T}) \setminus \Sigma)$ $\cap \mathsf{N_C}\rvert$	$\mathsf{Diff}_\Sigma(\mathrm{NCI}_{08.10e}, \mathrm{NCI}_{v08.09d})$ Success rate (%)	Avrg # Witnesses	$\mathsf{Diff}_\Sigma(\mathrm{NCI}_{05.05d}, \mathrm{NCI}_{v05.03d})$ Success rate (%)	Avrg # Witnesses	$\mathsf{Diff}_\Sigma(\mathrm{NCI}_{06.01c}, \mathrm{NCI}_{v05.12f})$ Success rate (%)	Avrg # Witnesses
5	98	2338.89	96	1347.92	99	13 704.29
10	98	2338.45	98	1348.47	100	13 788.15
20	97	2347.08	95	1348.66	95	13 841.52
50	92	2340.72	86	1351.56	87	14 062.52
100	86	2385.88	74	1354.04	80	14 504.40

Table 4: Computing the Logical Difference between Versions of NCI.

gradually. Figure 2 depicts the number of witnesses that correspond to the logical difference between consecutive versions of NCI on their common signature. Gonçalves, Parsia, and Sattler (2012b; 2012a) provide a comprehensive analysis of the changes between 14 consecutive versions of NCI using various techniques, ranging from a manual inspection of the log files to approximations of the logical difference. Versions 06.01c, 06.08d and 05.12f were identified as having the highest number of differences. In our experiments, the highest number of logical difference witnesses were also present in NCI version 06.01c; the computations for versions 06.08d and 05.12f did not finish in time.

Furthermore, to make the experiments more challenging for the reasoner, we focused on comparing version i with version $i+1$, and vice versa, on the 2 pairs of NCI versions for which the highest number of difference witnesses was identified in the first experiment. We also included version 08.10e as this is the last acyclic $\mathcal{ALC}$ TBox in the corpus. We performed tests on randomly generated large signatures Σ with $\Sigma \cap \mathsf{N_R} = \mathsf{sig}(\mathcal{T}) \cap \mathsf{N_R}$. In that way the computed uniform interpolants remained rather large as well.

For each sample size $x \in \{5, 10, 20, 50, 100\}$ we generated 100 signatures by randomly choosing $|\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_C}| - x$ concept names from $\mathsf{sig}(\mathcal{T})$ and by including all the role names from $\mathsf{sig}(\mathcal{T})$. The results that we obtained are now shown in Table 4.

One can observe that as size of $\mathsf{sig}(\mathcal{T}) \setminus \Sigma$ increased, i.e. more symbols had to be forgotten from the $\top\bot^*$ Σ-modules, the success rate dropped slightly. Overall, the average number of witnesses and the average maximal size of the witnesses remained comparable throughout the different sample sizes. Also, the axioms generated by the computation of the uniform interpolant did not pose a problem for FaCT++ as computing the logical difference for a given signature never took more than 20 seconds in our experiments.

Conclusion

In this paper we presented an approach based on clausal resolution for computing uniform interpolants of $\mathcal{ALC}$-TBoxes $\mathcal{T}$ w.r.t. signatures $\Sigma \subseteq \mathsf{sig}(\mathcal{T})$ that contain all the role names present in $\mathcal{T}$. We proved that whenever the saturation process under $\mathcal{ALC}$-resolution terminates, the algorithm computes a uniform interpolant. To guarantee termination on all inputs, we introduced a depth-bounded version of our algorithm. We showed that by choosing an appropriate bound on the depth of clauses, one can axiomatise all Σ-inclusions implied by the given TBox up to a specified depth. Combined with a known bound on the size of uniform interpolants, our depth-bounded procedure always computes a uniform interpolant if it exists.

In the second part of this paper we investigated how often our unrestricted resolution-based algorithm terminates with a uniform interpolant by applying our prototype implementation on a number of case studies. Our findings suggest that despite a high computational complexity uniform interpolants can be computed in many practical cases. The computation procedure could further benefit from better redundancy elimination techniques, which, together with extending our approach to forgetting role names, constitutes future work. It would also be interesting to explore proof strategies for our resolution calculi that guarantee termination when uniform interpolants exist.

Acknowledgements. Michel Ludwig is supported by the German Research Foundation (DFG) within the Cluster of Excellence 'Center for Advancing Electronics Dresden' (cfAED). Boris Konev is supported by the EPSRC project EP/H043594/1.

References

Auffray, Y.; Enjalbert, P.; and Hébrard, J.-J. 1990. Strategies for modal resolution: Results and problems. *Journal of Auto-*

mated Reasoning 6(1):1–38.

Baader, F.; Calvanese, D.; McGuinness, D. L.; Nardi, D.; and Patel-Schneider, P. F., eds. 2007. *The Description Logic Handbook*. Cambridge University Press, 2nd edition edition.

Baaz, M.; Egly, U.; and Leitsch, A. 2001. Normal form transformations. In *Handbook of Automated Reasoning*. Elsevier and MIT Press. 273–333.

Bachmair, L., and Ganzinger, H. 2001. Resolution theorem proving. In *Handbook of automated reasoning*, volume 1. Elsevier. chapter 2, 19–99.

Collberg, C. S.; Thomborson, C. D.; and Low, D. 1998. Manufacturing cheap, resilient, and stealthy opaque constructs. In *Proceedings of the 25th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL '98)*, 184–196. ACM.

Conradi, R., and Westfechtel, B. 1998. Version models for software configuration management. *ACM Computing Surveys (CSUR)* 30(2):232–282.

Cuenca Grau, B.; Horrocks, I.; Kazakov, Y.; and Sattler, U. 2008. Modular reuse of ontologies: theory and practice. *Journal of Artificial Intelligence Research (JAIR)* 31:273–318.

Eiter, T.; Ianni, G.; Schindlauer, R.; Tompits, H.; and Wang, K. 2006. Forgetting in managing rules and ontologies. In *Proceedings of the 2006 IEEE / WIC / ACM International Conference on Web Intelligence (WI 2006)*, 411–419. IEEE Computer Society.

Enjalbert, P., and del Cerro, L. F. 1989. Modal resolution in clausal form. *Theoretical Computer Science* 65(1):1–33.

Gonçalves, R. S.; Parsia, B.; and Sattler, U. 2012a. Concept-based semantic difference in expressive description logics. In *Description Logics*, volume 846 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Gonçalves, R. S.; Parsia, B.; and Sattler, U. 2012b. Concept-based semantic difference in expressive description logics. In *International Semantic Web Conference (1)*, volume 7649 of *Lecture Notes in Computer Science*, 99–115. Springer.

Herzig, A., and Mengin, J. 2008. Uniform interpolation by resolution in modal logic. In *Proceedings of the 11th European Conference on Logics in Artificial Intelligence (JELIA 2008)*, volume 5293 of *Lecture Notes in Computer Science*, 219–231. Springer.

Jiménez-Ruiz, E.; Grau, B. C.; Horrocks, I.; and Llavori, R. B. 2009. Contentcvs: A CVS-based collaborative ontology engineering tool. In *Proceedings of the Workshop on Semantic Web Applications and Tools for Life Sciences*, volume 559 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Jiménez-Ruiz, E.; Cuenca Grau, B.; Horrocks, I.; and Llavori, R. B. 2011. Supporting concurrent ontology development: Framework, algorithms and tool. *Data & Knowledge Engineering* 70(1):146–164.

Klein, M. C. A.; Fensel, D.; Kiryakov, A.; and Ognyanov, D. 2002. Ontology versioning and change detection on the web. In *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, volume 2473 of *Lecture Notes in Computer Science*. Berlin/Heidelberg, Germany: Springer Verlag. 247–259.

Konev, B.; Ludwig, M.; Walther, D.; and Wolter, F. 2012. The logical difference for the lightweight description logic $\mathcal{EL}$. *Journal of Artificial Intelligence Research (JAIR)* 44:633–708.

Konev, B.; Walther, D.; and Wolter, F. 2008. The logical difference problem for description logic terminologies. In *Proceedings of the 4th International Joint Conference on Automated Reasoning (IJCAR 2008)*, volume 5195 of *Lecture Notes in Computer Science*, 259–274. Springer.

Konev, B.; Walther, D.; and Wolter, F. 2009. Forgetting and uniform interpolation in large-scale description logic terminologies. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009)*, 830–835.

Koopmann, P., and Schmidt, R. A. 2013. Uniform interpolation of $\mathcal{ALC}$-ontologies using fixpoints. In *Proceedings of the 9th International Symposium on Frontiers of Combining Systems (FroCoS)*, volume 8152 of *Lecture Notes in Computer Science*. Springer.

Ludwig, M., and Konev, B. 2013. Towards practical uniform interpolation and forgetting for $\mathcal{ALC}$ TBoxes. In *Informal Proceedings of the 26th International Workshop on Description Logics*, volume 1014 of *CEUR Workshop Proceedings*, 377–389. CEUR-WS.org.

Lutz, C., and Wolter, F. 2011. Foundations for uniform interpolation and forgetting in expressive description logics. In Walsh, T., ed., *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011)*, 989–995.

Lutz, C.; Seylan, I.; and Wolter, F. 2012. An automata-theoretic approach to uniform interpolation and approximation in the description logic $\mathcal{EL}$. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning (KR 2012)*. AAAI Press.

Noy, N. F., and Musen, M. A. 2002. PromptDiff: A fixed-point algorithm for comparing ontology versions. In *Proceedings of the 18th national conference on Artificial intelligence*, 744–750. Menlo Park, CA, USA: AAAI Press.

Reiter, R., and Lin, F. 1994. Forget it! In *Proceedings of AAAI Fall Symposium on Relevance*, 154–159.

Sattler, U.; Schneider, T.; and Zakharyaschev, M. 2009. Which kind of module should I extract? In *Proceedings of the 22nd International Workshop on Description Logics (DL 2009)*, volume 477 of *CEUR Workshop Proceedings*. CEUR-WS.org.

ten Cate, B.; Conradie, W.; Marx, M.; and Venema, Y. 2006. Definitorially complete description logics. In *Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning (KR 2006)*, 79–89. AAAI Press.

Tsarkov, D., and Horrocks, I. 2006. FaCT++ Description Logic reasoner: System description. In *Proceedings of the Third International Joint Conference on Automated Reason-*

ing (IJCAR 2006), volume 4130 of *Lecture Notes in Computer Science*, 292–297. Springer.

Wang, Z.; Wang, K.; Topor, R.; and Pan, J. Z. 2008. Forgetting Concepts in DL-Lite. In *Proceedings of the 5th European Semantic Web Conference (ESWC2008)*, volume 5021 of *Lecture Notes in Computer Science*, 245–257. Springer.

Wang, Z.; Wang, K.; Topor, R. W.; and Zhang, X. 2010. Tableau-based forgetting in $\mathcal{ALC}$ ontologies. In *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI 2010)*, volume 215 of *Frontiers in Artificial Intelligence and Applications*, 47–52.

Wang, K.; Wang, Z.; Topor, R.; Pan, J. Z.; and Antoniou, G. 2012. Eliminating concepts and roles from ontologies in expressive descriptive logics. *Computational Intelligence*. DOI: 10.1111/j.1467-8640.2012.00442.x.

Zhou, Y., and Zhang, Y. 2011. Bounded forgetting. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence (AAAI 2011)*. AAAI Press.

The appendix is organised as follows. In Section B we give a proof of the refutational completeness with ordering constraints for the calculus $\mathfrak{T}$. We continue in Section C with establishing an important connection between the calculi $\mathfrak{T}$ and $\mathfrak{T}^u$, which leads to a proof of the refutational completeness for $\mathfrak{T}^u$. The aim of Section D is to establish the correctness of Algorithm 1, which includes a proof the subsumption lemma for $\mathfrak{T}^f$ as well as some results regarding the application of the Supp-operator. The correctness of the depth-bounded uniform interpolation algorithm is established in the sections E and F for the two cases of regular and acyclic $\mathcal{ALC}$-TBoxes. In Section G we address the correctness of the implementation by extending the subsumption lemma to $\mathfrak{T}^{u,f}$.

A Extended Definitions

In the following let $\mathcal{R}$ denote the set of all sequences $\mathbf{r} = (r_1, \ldots, r_n)$ with $r_i \in \mathsf{N_R}$ for every $1 \leq i \leq n$. The empty sequence will be denoted by ϵ.

For $\mathbf{r} = (r_1, \ldots, r_n) \in \mathcal{R}$, the expression $\forall \mathbf{r}.\mathcal{C}$, where $\mathcal{C}$ is a clause, denotes the clause $\forall r_1. \ldots \forall r_n.\mathcal{C}$.

The proof of the following lemma is a simple adaptation of CNF transformation for first-order formulae (Baaz, Egly, and Leitsch 2001).

Lemma 14 (Clausification)**.** *Let C be an $\mathcal{ALC}$-concept. Then one can construct a set of clauses $\mathrm{Cls}(C)$ from C such that*

- $\mathsf{sig}(\mathrm{Cls}(C)) \subseteq \mathsf{sig}(C)$,
- $\max\{\, \mathrm{Depth}(\mathcal{C}) \mid \mathcal{C} \in \mathrm{Cls}(C)\,\} \leq \mathrm{Depth}(C)$, *and*
- $\models C \equiv \bigsqcap_{\mathcal{C} \in \mathrm{Cls}(C)} \mathcal{C}$.

The set of all clauses over $\mathsf{N_C} \cup \mathsf{N_R}$ will be denoted by Clauses, whereas $\mathrm{Clauses}_n(\Upsilon)$ denotes the set of all clauses over the signature Υ which are of depth at most n. For a TBox $\mathcal{T}$ we define that

$$\begin{aligned}\mathrm{Cls}(\mathcal{T}) = \{\, \mathrm{Cls}(\neg C \sqcup D) \mid C \sqsubseteq D \in \mathcal{T}\,\} &\cup \{\, \mathrm{Cls}(\neg C \sqcup D) \mid C \equiv D \in \mathcal{T}\,\} \\ &\cup \{\, \mathrm{Cls}(\neg D \sqcup C) \mid C \equiv D \in \mathcal{T}\,\}.\end{aligned}$$

For a clause $\mathcal{C}$ and $r \in \mathsf{sig}(\mathcal{C})$, we define $\mathcal{C}|_{\exists r}$ ($\mathcal{C}|_{\forall r}$) to be the set of all the literals of the form $\exists r.\mathcal{F}$ ($\forall r.\mathcal{D}$) contained in $\mathcal{C}$. Furthermore, for a clause $\mathcal{C}$ let $\mathcal{C}|_{\exists} = \bigcup_{r \in \mathsf{sig}(\mathcal{C})} \mathcal{C}|_{\exists r}$ and $\mathcal{C}|_{\forall} = \bigcup_{r \in \mathsf{sig}(\mathcal{C})} \mathcal{C}|_{\forall r}$.

We define the set of subconcepts of a clause $\mathcal{C}$, $\mathsf{sub}(\mathcal{C})$, (or of a set of clauses $\mathcal{F}$) inductively as follows:

- for a clause $\mathcal{C}$, let
$$\mathsf{sub}(\mathcal{C}) = \{\mathcal{C}\} \cup \mathcal{C} \cup \bigcup_{\exists r.\mathcal{F} \in \mathcal{C}} \mathsf{sub}(\mathcal{F}) \cup \bigcup_{\forall r.\mathcal{D} \in \mathcal{C}} \mathsf{sub}(\mathcal{D})$$
- for a set of clauses $\mathcal{F}$, let $\mathsf{sub}(\mathcal{F}) = \bigcup_{\mathcal{C} \in \mathcal{F}} \mathsf{sub}(\mathcal{C})$.

In order to be able to prove completeness and reordering theorems, we need to be able to refer to individual inference rules of $\mathfrak{T}$ and $\mathfrak{T}^u$ by their name. We re-state the rules of the calculus $\mathfrak{T}$ and $\mathfrak{T}^u$. Then, $\mathfrak{T}$ consists of two rules

$$\frac{\mathcal{C}}{\mathcal{E}_1}\ (i_1) \qquad\qquad \frac{\mathcal{C} \quad \mathcal{D}}{\mathcal{E}_2}\ (i_2)$$

where $\mathcal{C}, \mathcal{D}$ are initial clauses, $\mathcal{C} \Rightarrow_\alpha \mathcal{E}_1$, and $\mathcal{C}, \mathcal{D} \Rightarrow_\alpha \mathcal{E}_2$. $\mathcal{E}_1$ and $\mathcal{E}_2$ are initial clauses.

The calculus $\mathfrak{T}^u$ consists of two rules (i_1) and (i_2), and the following three inference rules:

$$\frac{\mathcal{C}}{\mathcal{E}_1}\ (u_1) \qquad\qquad \frac{\mathcal{C} \quad \mathcal{D}}{\mathcal{E}_2}\ (u_2) \qquad\qquad \frac{\mathcal{C}' \quad \mathcal{D}}{\mathcal{E}'}\ (\mathrm{mix})$$

where $\mathcal{C}, \mathcal{D}$ are universal clauses, $\mathcal{C}'$ is an initial clause or a universal clause, $\mathcal{C} \Rightarrow_\alpha \mathcal{E}_1, \mathcal{C}, \mathcal{D} \Rightarrow_\alpha \mathcal{E}_2$, and there exists $\mathbf{r} \in \mathcal{R} \setminus \{\epsilon\}$ such that $\mathcal{C}', \forall \mathbf{r}.\mathcal{D} \Rightarrow_\alpha \mathcal{E}'$. $\mathcal{E}_1$ and $\mathcal{E}_2$ are universal clauses, whereas $\mathcal{E}'$ is a universal clause if $\mathcal{C}'$ is a universal clause and $\mathcal{E}'$ is an initial clause otherwise. We also assume that every clause $\mathcal{E}$ that results from a $\mathfrak{T}$- or $\mathfrak{T}^u$-inference is implicitly simplified by exhaustively removing all occurrences of literals of the form $\exists r.(\mathcal{F}, \bot)$.

In the following we also make use of two additional calculi $\mathfrak{T}^f$ and $\mathfrak{T}^{u,f}$, which are defined analogously to the calculi $\mathfrak{T}$ and $\mathfrak{T}^u$, respectively, except that they are based on the inference relation $\Rightarrow^f_\alpha$.

For $x \in \{\mathfrak{T}, \mathfrak{T}^f, \mathfrak{T}^u, \mathfrak{T}^{u,f}\}$ and $\alpha \in \mathsf{N_C} \cup \{\bot, \exists_f\}$, we write $\mathcal{C}(,\mathcal{D}) \vdash^\alpha_x \mathcal{E}$ if, and only if, the clause $\mathcal{E}$ results from an x-inference on α from $\mathcal{C}$ (and $\mathcal{D}$).

Definition 15. *Let Υ be a signature and let $\mathcal{N}$ be a set of $\mathcal{ALC}$ clauses. For $x \in \{\mathfrak{T}, \mathfrak{T}^u\}$, we define that*

$$\begin{aligned}\mathrm{Res}_{x,\Upsilon,m}(\mathcal{N}) = \mathcal{N} &\cup \{\, \mathcal{C}' \mid \mathcal{C}_1, \mathcal{C}_2 \in \mathcal{N}, \mathcal{C}_1, \mathcal{C}_2 \vdash^\alpha_x \mathcal{C}', \alpha \in \Upsilon \textit{ and } \mathrm{Depth}(\mathcal{C}') \leq m\,\} \\ &\cup \{\, \mathcal{C}' \mid \mathcal{C}_1 \in \mathcal{N}, \mathcal{C}_1 \vdash^\alpha_x \mathcal{C}', \alpha \in \Upsilon \textit{ and } \mathrm{Depth}(\mathcal{C}') \leq m\,\}.\end{aligned}$$

Additionally, for $n \in \mathbb{N} \setminus \{0\}$, we define that $\mathrm{Res}^0_{x,\Upsilon,m}(\mathcal{N}) = \mathcal{N}$, and $\mathrm{Res}^n_{x,\Upsilon,m}(\mathcal{N}) = \mathrm{Res}_{x,\Upsilon,m}(\mathrm{Res}^{n-1}_{x,\Upsilon,m}(\mathcal{N}))$. Finally, we set

$$\mathrm{Res}^\infty_{x,\Upsilon,m}(\mathcal{N}) = \bigcup_{n \in \mathbb{N}\setminus\{0\}} \mathrm{Res}^{n-1}_{x,\Upsilon,m}(\mathcal{N}).$$

Definition 16 (Derivation Tree)**.** *Let $x \in \{\mathfrak{I}, \mathfrak{I}^u\}$. A x-derivation tree Δ built from a set of clauses $\mathcal{N}$ is a finite binary tree $\Delta = (V, E, L)$ where L is a labelling function which assigns to every node n an $\mathcal{ALC}$ clause $L(n)$ such that*

(i) the leaves of Δ are only assigned clauses from $\mathcal{N}$; and

(ii) for every node $n \in V$ with immediate predecessors n_1 (and potentially n_2) it holds that $L(n)$ is the result of an inference rule from x applied on $L(n_1)$ and potentially $L(n_2)$.

An inference path *of Δ is a sequence $(\alpha_2, \dots, \alpha_m)$ where $(n_1, \dots, n_m)$ is a path from a leaf n_1 of Δ to the root n_m and for every $2 \leq i \leq m$, the clause $L(n_i)$ has been obtained through an $\Rightarrow_{\alpha_i}$ inference. The set of all inferences paths of Δ is denoted by* $\mathrm{InferencePath}(\Delta)$*. The set of all inferences that occur in Δ, which is denoted by* $\mathrm{Inferences}(\Delta)$*, is defined as follows:*

$$\mathrm{Inferences}(\Delta) = \bigcup_{p=\alpha_2,\dots,\alpha_m \in \mathrm{InferencePath}(\Delta)} \{\, \alpha_i \mid 2 \leq i \leq m \,\}$$

We will denote the set which contains all the leaves of Δ by $\mathrm{Leaves}(\Delta)$*, and we set* $\mathrm{Nodes}(\Delta) = V$*, Clauses*$(\Delta) = \{\, L(n) \mid n \in \mathrm{Nodes}(\Delta) \,\}$.

Definition 17 ($(\Upsilon, \succ)$-Path)**.** *A sequence $(\alpha_1, \dots, \alpha_m)$ over $(\mathsf{N_C} \cup \{\bot\})$ is a $(\Upsilon, \succ)$-path if, and only if,*

- *there exists $0 \leq i \leq m$ such that $\alpha_j \in \Upsilon$ for every $1 \leq j \leq i$ and $\alpha_i \succeq \alpha_{i+1}$ for every $1 \leq j < i$; and*
- *$\alpha_j \notin \Upsilon$ for every $i + 1 \leq j \leq m$.*

Definition 18 (Υ-Derivation Tree)**.** *Let $\Upsilon \subseteq \mathsf{N_C}$ be a set of concept names, let $\mathcal{N}$ be a set of $\mathcal{ALC}$ clauses, and let $x \in \{\mathfrak{I}, \mathfrak{I}^u\}$.*

A $(x, \Upsilon, \succ)$-derivation tree Δ built from the set $\mathcal{N}$ is a x-derivation tree $\Delta = (V, E)$ such that every path $(\alpha_1, \dots, \alpha_m) \in$ $\mathrm{InferencePath}(\Delta)$ *is a $(\Upsilon, \succ)$-path.*

B Proof of Refutational Completeness for $\mathfrak{I}$ with Ordering Constraints on Inferences

In this section we prove refutational completeness of $\mathfrak{I}$ with ordering constraints. A weaker formulation of this result has already been announced in (Herzig and Mengin 2008); however, the proof given in (Herzig and Mengin 2008) does not seem to be correct. A pivotal for the method of (Herzig and Mengin 2008) is the following statement.

CLAIM. [Lemma 1 of (Herzig and Mengin 2008)] Given clauses $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{I}$ and $\mathcal{F}$, if there is an α-resolution from $\mathcal{A}$, possibly using side clause $\mathcal{B}$, giving clause $\mathcal{I}$ and a β-resolution from $\mathcal{I}$, possibly using side clause $\mathcal{C}$, giving clause $\mathcal{F}$, then there exist clauses $\mathcal{I}^*$, $\mathcal{F}^*_1$,…, $\mathcal{F}^*_n$ (for some $n \geq 0$) and $\mathcal{F}^*$ such that $\mathcal{F}^*$ subsumes $\mathcal{F}$ and there is a β-resolution from $\mathcal{A}$, possibly using side clause $\mathcal{C}$, giving clause $\mathcal{I}^*$, and a sequence of α-resolutions from $\mathcal{I}^*$, possibly using side clause $\mathcal{B}$, giving successively $\mathcal{F}^*_1$,…, $\mathcal{F}^*_N$, $\mathcal{F}^*$.

To observe where the claim falls short consider a clause set $S = \{\forall r.A, \forall r.\neg A, \exists r.C\}$. Then from $\forall r.A$ and $\forall r.\neg A$ we can derive $\forall r.\bot$, and then $\bot$ using $\exists r.C$, i.e. we first resolved on A and then on $\bot$. However, contrary to the claim above, it's not possible to first resolve on $\bot$ using clauses from S only. As the following (more involved) example demonstrates, the problem is not limited to β being $\bot$.

Example 19. *Consider a set of clauses $S' = \{\mathcal{C}_1 = \neg A \sqcup B, \mathcal{C}_2 = A \sqcup B, \mathcal{C}_3 = \neg B\}$. By first resolving on A between $\mathcal{C}_1$ and $\mathcal{C}_2$, one obtains the unit clause B, from which one can derive the empty clause by resolving on B with $\mathcal{C}_3$. But by first resolving on B between $\mathcal{C}_1$ and $\mathcal{C}_3$ one obtains the unit clause $\neg A$, which can be resolved with $\mathcal{C}_2$ on A to derive B. But no further inferences on A are possible and $B \leq_s \bot$ does not hold. (This problem is caused by the implicit positive factorisation built into the calculus of (Enjalbert and del Cerro 1989) and (Herzig and Mengin 2008).)*

Additionally, for the proof of Proposition 1 in (Herzig and Mengin 2008) to work, it would be necessary to have proved the subsumption lemma already.

In our approach rather than restructuring a given proof we impose ordering constraints on derivations directly in the proof of completeness. Following (Enjalbert and del Cerro 1989), we start by introducing a variation of tableau for $\mathcal{ALC}$ clauses and then show that an ordered derivation can be effectively constructed from the tableau.

Definition 20 (Model Tree)**.** *Let $\mathcal{T}$ be a TBox and let $\mathcal{N}$ be a finite set of $\mathcal{ALC}$ clauses.*

A model tree *$\mathcal{M} = (V, E, L)$ for the TBox $\mathcal{T}$ and the set of clauses $\mathcal{N}$ is a directed and labelled tree (V, E) with a node labelling function $L\colon V \to \wp(\text{Clauses})$ that is constructed iteratively as follows from $V = E = \emptyset$:*

- *Add a root node n_0 to V with $L(n_0) = \mathcal{N} \cup \mathrm{Cls}(\mathcal{T})$*

- *Additional children are constructed by alternating between the following two operations on leaf nodes n:*

  Operation 1: Repeat as long as possible

  (i) choose a leaf n and a clause $\mathcal{C}$ in $L(n)$ of the form $\mathcal{C}_1 \sqcup \mathcal{C}_2$ with $\mathcal{C}_1 \neq \emptyset$ and $\mathcal{C}_2 \neq \emptyset$

  (ii) append two children n_1, n_2 to n such that:

$$L(n_1) = n \setminus \{\mathcal{C}\} \cup \{\mathcal{C}_1\} \text{ and } L(n_2) = n \setminus \{\mathcal{C}\} \cup \{\mathcal{C}_2\}$$

  Operation 2: For each leaf node n of $\mathcal{M}$,

- *if $\{A, \neg A\} \subseteq L(n)$ for some concept name A, or $\bot \in L(n)$, or if there exists an ancestor n' with $L(n') = L(n)$, then do nothing;*
- *otherwise, as $L(n)$ is a set of unit clauses, we can write*

$$L(n) = \{\mathcal{L}_1, \ldots, \mathcal{L}_m\} \cup \bigcup_{i=1}^{l} \{\exists r_i.\mathcal{F}_i\} \cup \bigcup_{j=1}^{p} \{\forall s_j.\mathcal{C}_j\},$$

  where every $\mathcal{L}_i$, $1 \leq i \leq m$ is either a concept names of the negation of a concept names.
  Define children n_k ($1 \leq k \leq l$) of n such that:

$$L(n_k) = \mathcal{F}_k \cup \{\, \mathcal{C}_j \mid 1 \leq j \leq p \text{ and } r_k = s_j \,\} \cup \mathrm{Cls}(\mathcal{T})$$

  *The node n_k is said to be an r_k-*successor *of n.*

Remark 21. *As $L(n) \subseteq \mathsf{sub}(\mathcal{N} \cup \mathrm{Cls}(\mathcal{T}))$ holds for $n = n_0$ and for every node $n \in V$ that results from Operation 2, and as $\mathsf{sub}(\mathcal{N} \cup \mathrm{Cls}(\mathcal{T}))$ is finite, one can see that every model tree for $\mathcal{N}$ w.r.t. $\mathcal{T}$ only contains finitely many nodes.*

Lemma 22. *In a model tree $\mathcal{M}$ for $\mathcal{N}$ w.r.t. $\mathcal{T}$ a node that contains A, $\neg A$ for some concept name A cannot have a type 2 descendant.*

Proof. Follows immediately from the construction principles of model trees. □

Definition 23. *In a model tree $\mathcal{M}$ for $\mathcal{N}$ w.r.t. $\mathcal{T}$*

- *a leaf node n is said to be* closed *if and only if $\{A, \neg A\} \subseteq L(n)$ for some concept name A, or $\bot \in L(n)$;*
- *for a node n of type 1 with successors, n is said to be closed if and only if all of its children are closed;*
- *for a node n of type 2 with successors, n is said to be closed if and only if one of its children n' is closed.*

A node n which is not closed is said to be open*.*

The following lemma can easily be proved by induction on the size of a model tree.

Lemma 24. *Let n be an open node in a model tree $\mathcal{M}$ for $\mathcal{N}$ w.r.t. $\mathcal{T}$.*
Then there exists a sub-tree S of $\mathcal{M}$ with root n such that

- *every node of S is open, and*
- *every node of type 1 has exactly one child, and*
- *the children of every node n' of type 2 in S are exactly the children of n' in $\mathcal{M}$ of $\mathcal{N}$ w.r.t $\mathcal{T}$.*

The following proofs are variations/extensions of the proofs found in (Enjalbert and del Cerro 1989).

Lemma 25. *Let $\mathcal{T}$ be a TBox and let $\mathcal{N}$ be a finite set of clauses.*
Then for every open model tree $\mathcal{M}$ for $\mathcal{N}$ w.r.t. $\mathcal{T}$ there exists a model $\mathcal{I}$ of $\mathcal{T}$ such that $\bigcap_{\mathcal{C} \in \mathcal{N}} \mathcal{C}^{\mathcal{I}} \not\subseteq \bot^{\mathcal{I}}$.

Proof. Let $\mathcal{S} = (V, E, L)$ be the sub-tree of $\mathcal{M}$ for $\mathcal{N}$ w.r.t $\mathcal{T}$, with root n_0 that is obtained from Lemma 24. Additionally, let $\mathfrak{R}$ be the smallest equivalence relation containing (n, n') for every node n of $\mathcal{S}$ such that n' is a type 1-child of n. The equivalence class of a node $n \in \mathcal{S}$ w.r.t. $\mathfrak{R}$ will be denoted by $|n|$.

We define the interpretation $\mathcal{I}$ as follows:

- $\Delta = \{\, |n| \mid n \in E \,\}$,
- for every atomic concept name A:

$$A^{\mathcal{I}} = \{\, |n| \mid \exists\, n' \in |n| \text{ such that } A \in L(n') \,\},$$

- for every role name r:

$$\begin{aligned} r^{\mathcal{I}} =& \{\, (|n_1|, |n_2|) \mid (n_1, n_2) \in \mathcal{S} \text{ and } n_2 \text{ is an } r\text{-successor of } n_1 \,\} \\ & \cup \{\, (|n_1|, |n_2|) \mid \exists\, \tilde{n}_1 \text{ such that } \tilde{n}_1 \text{ is an ancestor of } n_1, \\ & L(n_1) = L(\tilde{n}_1) \text{ and } \tilde{n}_2 \text{ is an } r\text{-successor of } \tilde{n}_1 \,\} \end{aligned}$$

We now prove for every clause $\mathcal{C}$ and for every node n of S with $\mathcal{C} \in L(n)$ that $|n| \in \mathcal{C}^{\mathcal{I}}$ holds by induction on the structure of $\mathcal{C}$.

If $\mathcal{C} = A$, then $|n| \in A^{\mathcal{I}}$ holds for every node n of S with $\mathcal{C} \in L(n)$ by definition of the interpretation $\mathcal{I}$.

For $\mathcal{C} = \neg A$, let n be a node of S with $\mathcal{C} \in L(n)$. It follows for every node $n' \in |n|$ that $A \notin n'$ as every such node n' is open and as the presence of A in a node n' of type 1 would imply that A is also present in all of its type 1-children. Hence, we can infer that $|n| \notin A^{\mathcal{I}}$, i.e. $|n| \in (\neg A)^{\mathcal{I}}$ holds.

In the case where $\mathcal{C} = \mathcal{C}_1 \sqcup \mathcal{C}_2$, let n be a node of S with $\mathcal{C} \in L(n)$. We can infer that the node n is of type 1, i.e. there exists a type 1 descendant n' of $\mathcal{C}$ such that either $\mathcal{C}_1 \in L(n')$ or $\mathcal{C}_2 \in L(n')$. It follows from the induction hypothesis that $|n'| \in (\mathcal{C}_1)^{\mathcal{I}}$, or $|n'| \in (\mathcal{C}_2)^{\mathcal{I}}$. We can conclude that $|n| = |n'| \in \mathcal{C}^{\mathcal{I}}$.

For $\mathcal{C} = \exists r.\mathcal{F}$, let n be a node of S with $\mathcal{C} \in L(n)$. By construction of $\mathcal{S}$ there exists a node $n' \in |n|$ such that $\exists r.\mathcal{F} \in L(n')$ and such that all the clauses in $L(n')$ are unit clauses. As n' is open, there hence exist nodes $\tilde{n}', \tilde{n}'' \in S$ such that $L(n') \subseteq L(\tilde{n}')$ and $\tilde{n}''$ is an r-successor of $\tilde{n}'$ with $\mathcal{F} \subseteq L(n'')$, i.e. either $n' = \tilde{n}'$ or $\tilde{n}'$ is an ancestor of n' with $L(n) = L(n')$. It then follows from the induction hypothesis that $|\tilde{n}''| \in (\bigsqcap_{\mathcal{D} \in \mathcal{F}} \mathcal{D})^{\mathcal{I}}$. As $(|n'|, |\tilde{n}''|) \in r^{\mathcal{I}}$, it follows that $|n| = |n'| \in \mathcal{C}^{\mathcal{I}}$.

Finally, if $\mathcal{C} = \forall r.\mathcal{D}$, let n be a node of S with $\mathcal{C} \in L(n)$ and let $|n'| \in \Delta$ such that $(|n|, |n'|) \in r^{\mathcal{I}}$. By construction of $\mathcal{S}$ there exists a node $n_u \in |n|$ such that $\forall r.\mathcal{D} \in L(n_u)$ and such that all the clauses in $L(n_u)$ are unit clauses. By definition of $r^{\mathcal{I}}$ there hence exists a node $\tilde{n}$ in S such that $L(n_u) = L(\tilde{n})$ and such that there exists a node $\tilde{n}'$ which is a r-successor of $\tilde{n}$ with $|\tilde{n}'| = |n'|$, i.e. we have $\mathcal{D} \in \tilde{n}'$ by construction of $\mathcal{S}$. Thus, $|n'| = |\tilde{n}'| \in \mathcal{D}^{\mathcal{I}}$ by the induction hypothesis. We can hence infer that $|n| \in \mathcal{C}^{\mathcal{I}}$ holds.

We can conclude now that $\mathcal{I}$ is a model of $\mathcal{T}$ as for every $|n| \in \Delta^{\mathcal{I}}$ there exists $\tilde{n} \in |n|$ with $\mathrm{Cls}(\mathcal{T}) \subseteq L(\tilde{n})$, which implies that $|n| = |\tilde{n}| \in \bigcap_{\mathcal{D} \in \mathrm{Cls}(\mathcal{T})} \mathcal{D}^{\mathcal{I}}$. Finally, it remains to observe that as $\mathcal{N} \subseteq L(n_0)$, we have $|n_0| \in \bigcap_{\mathcal{C} \in \mathcal{N}} \mathcal{C}^{\mathcal{I}}$ and therefore, $\bigcap_{\mathcal{C} \in \mathcal{N}} \mathcal{C}^{\mathcal{I}} \nsubseteq \perp^{\mathcal{I}}$. □

We summarise the properties of the model trees in the following two lemmas, which can easily be proved by induction on the construction of the model tree.

Lemma 26. *Let $\mathcal{T}$ be a TBox, let $\mathcal{N}$ be a finite set of clauses and let $\mathcal{I}$ be a model of $\mathcal{T}$ such that $\bigcap_{\mathcal{C} \in \mathcal{N}} \mathcal{C}^{\mathcal{I}} \nsubseteq \perp^{\mathcal{I}}$.*
Then there exists an open model tree $\mathcal{M}$ for $\mathcal{N}$ w.r.t. $\mathcal{T}$.

Lemma 27. *Let n be a closed node in a model tree $\mathcal{M}$ for $\mathcal{N}$ w.r.t. $\mathcal{T}$.*
Then there exists a finite tree T with root n which is a sub-tree of $\mathcal{M}$ and which only contains closed nodes.

Our next aim is to show that one can construct a $(\mathfrak{T}, \Upsilon, \succ)$-derivation out of the clauses contained in an arbitrary closed node in a model tree. We proceed by induction on the depth of a closed node. In order to do that we distinguish between whether the considered close node results from type 1 or type 2 expansion of the model tree, and we begin with showing that refutations can be constructed for closed type 1 nodes. The main result will be established in Lemma 41 by using the property that inferences can be reordered (Lemma 33). However, to avoid complications that can arise from implicit factoring as illustrated in Example 19 above we 'split' clauses using the following notion of a bipartite derivation. Then, when dealing with closed type 1 nodes and the associated unsatisfiable disjunction of the form $\mathcal{D}_1 \sqcup \mathcal{D}_2$, refutations for $\mathcal{D}_i$, for $i = 1, 2$ belong to different partitions and can, therefore, be treated independently.

Definition 28. *For a clause $\mathcal{C}$ and for $x \in \{l, r\}$ we denote by $[\mathcal{C}]_x$ the clause that results from $\mathcal{C}$ by consistently replacing every concept name A which occurs in $\mathcal{C}$ with the concept name A_x and every role name r which occurs in $\mathcal{C}$ with r_x. We assume that A clause $\mathcal{C}$ is said to be* bipartite *if, and only if, there exist clauses $\mathcal{D}$ and $\mathcal{E}$ such that $\mathcal{C} = [\mathcal{D}]_l \cup [\mathcal{E}]_r$. A bipartite clause $\mathcal{C}$ is also denoted by $[\mathcal{C}]$.*
For a bipartite clause $\mathcal{C} = [\mathcal{D}]_l \cup [\mathcal{E}]_r$, $[\mathcal{C}]_l$ denotes the clause $\mathcal{D}$ and $[\mathcal{C}]_r$ denotes the clause $\mathcal{E}$.

Definition 29. *A $\mathfrak{T}$-derivation Δ from premises $\mathcal{N}$ is said to be* bipartite *if, and only if, the clauses in $\mathcal{N}$ are bipartite.*

We establish properties of bipartite derivations. The following statement is a direct consequence of the bipartite derivation definition.

Lemma 30. *Let Δ be a bipartite derivation. Then it holds that every clause that occurs in Δ is bipartite and for every derivation step $[\mathcal{C}](, [\mathcal{D}]) \Rightarrow_\alpha [\mathcal{E}]$ in Δ it holds that either*

- $[\mathcal{C}]_l(, [\mathcal{D}]_l) \Rightarrow_\alpha [\mathcal{E}]_l$ *and* $[\mathcal{E}]_r = [\mathcal{C}]_r \cup [\mathcal{D}]_r$, *or*
- $[\mathcal{E}]_l = [\mathcal{C}]_l \cup [\mathcal{D}]_l$ *and* $[\mathcal{C}]_r(, [\mathcal{D}]_r) \Rightarrow_\alpha [\mathcal{E}]_r$.

Lemma 31. *If $[\mathcal{C}_1]_l \cup [\mathcal{C}_2]_r(, [\mathcal{D}_1]_l \cup [\mathcal{D}_2]_r) \Rightarrow_{[\alpha]_x} [\mathcal{E}_1]_l \cup [\mathcal{E}_2]_r$, where $x \in \{l, r\}$, and $[\mathcal{C}_1']_l \cup [\mathcal{C}_2']_r \subseteq [\mathcal{C}_1]_l \cup [\mathcal{C}_2]_r$, $[\mathcal{D}_1']_l \cup [\mathcal{D}_2']_r \subseteq [\mathcal{D}_1]_l \cup [\mathcal{D}_2]_r$, then either*

- $[\mathcal{C}_1']_l \cup [\mathcal{C}_2']_r \subseteq [\mathcal{E}_1]_l \cup [\mathcal{E}_2]_r$, *or*
- $[\mathcal{D}_1']_l \cup [\mathcal{D}_2']_r \subseteq [\mathcal{E}_1]_l \cup [\mathcal{E}_2]_r$, *or*
- $[\mathcal{C}_1']_l \cup [\mathcal{C}_2']_r(, [\mathcal{D}_1']_l \cup [\mathcal{D}_2']_r) \Rightarrow_{[\alpha]_x} [\mathcal{E}_1']_l \cup [\mathcal{E}_2']_r$ *such that* $[\mathcal{E}_1']_l \cup [\mathcal{E}_2']_r \subseteq [\mathcal{E}_1]_l \cup [\mathcal{E}_2]_r$.

Proof. We distinguish between the following two cases.

If $\mathcal{C}_1(,\mathcal{D}_1) \Rightarrow_\alpha \mathcal{E}_1$, let $\mathcal{L}_{\mathcal{C}_1}$ and $\mathcal{L}_{\mathcal{D}_1}$ be the respective literals of $\mathcal{C}_1$ and $\mathcal{D}_1$ that are resolved upon, i.e. $\mathcal{L}_{\mathcal{C}_1} \in \mathcal{C}_1$, $\mathcal{L}_{\mathcal{D}_1} \in \mathcal{D}_1$, and $\mathcal{E}_1 = \mathcal{C}_1 \setminus \{\mathcal{L}_{\mathcal{C}_1}\} \sqcup \mathcal{D}_1 \setminus \{\mathcal{L}_{\mathcal{D}_1}\} \sqcup \mathcal{L}$ with $\mathcal{L}_{\mathcal{C}_1}(,\mathcal{L}_{\mathcal{D}_1}) \Rightarrow_\alpha \mathcal{L}$ ($\mathcal{L}$ may be $\bot$). It also holds that $\mathcal{E}_2 = \mathcal{C}_2 \sqcup \mathcal{D}_2$. Now, if $\mathcal{L}_{\mathcal{C}_1} \notin \mathcal{C}'_1$, then $\mathcal{C}'_1 \subseteq \mathcal{C}_1 \setminus \{\mathcal{L}_{\mathcal{C}_1}\}$ and $[\mathcal{C}'_1]_l \cup [\mathcal{C}'_2]_r \subseteq [\mathcal{E}_1]_l \cup [\mathcal{E}_2]_r$ holds. Similarly, if $\mathcal{L}_{\mathcal{D}_1} \notin \mathcal{D}'_1$, we have $\mathcal{D}'_1 \subseteq \mathcal{D}_1 \setminus \{\mathcal{L}_{\mathcal{D}_1}\}$ and $[\mathcal{D}'_1]_l \cup [\mathcal{D}'_2]_r \subseteq [\mathcal{E}_1]_l \cup [\mathcal{E}_2]_r$. We can now assume that $\mathcal{L}_{\mathcal{C}_1} \in \mathcal{C}'_1$ and $\mathcal{L}_{\mathcal{D}_1} \in \mathcal{D}'_1$. It is then easy to see that $\mathcal{C}'_1(,\mathcal{D}'_1) \Rightarrow_\alpha \mathcal{E}'_1$ with $\mathcal{E}'_1 = \mathcal{C}'_1 \setminus \{\mathcal{L}_{\mathcal{C}_1}\} \sqcup \mathcal{D}'_1 \setminus \{\mathcal{L}_{\mathcal{D}_1}\} \sqcup \mathcal{L}$ and $\mathcal{E}'_1 \subseteq \mathcal{E}_1$. We can conclude that $[\mathcal{E}'_1]_l \cup [\mathcal{E}'_2]_r \subseteq [\mathcal{E}_1]_l \cup [\mathcal{E}_2]_r$ holds as $\mathcal{E}'_2 = \mathcal{C}'_2 \cup \mathcal{D}'_2 \subseteq \mathcal{C}_2 \cup \mathcal{D}_2 = \mathcal{E}_2$.

The case for $\mathcal{C}_2(,\mathcal{D}_2) \Rightarrow_\alpha \mathcal{E}_2$ can be proved analogously. □

Lemma 32. *If $[\mathcal{C}_1]_l \cup [\mathcal{C}_2]_r, [\mathcal{D}_1]_l \cup [\mathcal{D}_2]_r \Rightarrow_{[\alpha]_x} [\mathcal{E}_1]_l \cup [\mathcal{E}_2]_r$ and $\mathcal{C}^* \subseteq \mathcal{C}_1 \cup \mathcal{C}_2$, $\mathcal{D}^* \subseteq \mathcal{D}_1 \cup \mathcal{D}_2$, then either*

- *$\mathcal{C}^* \subseteq \mathcal{E}_1 \cup \mathcal{E}_2$, or*
- *$\mathcal{D}^* \subseteq \mathcal{E}_1 \cup \mathcal{E}_2$, or*
- *$\mathcal{C}^*, (\mathcal{D}^*) \Rightarrow_\alpha \mathcal{E}^*$ such that $\mathcal{E}^* \subseteq \mathcal{E}_1 \cup \mathcal{E}_2$.*

Proof. We distinguish between the following two cases.

If $\mathcal{C}_1(,\mathcal{D}_1) \Rightarrow_\alpha \mathcal{E}_1$, let $\mathcal{L}_{\mathcal{C}_1}$ and $\mathcal{L}_{\mathcal{D}_1}$ be the respective literals of $\mathcal{C}_1$ and $\mathcal{D}_1$ that are resolved upon, i.e. $\mathcal{E}_1 = \mathcal{C}_1 \setminus \{\mathcal{L}_{\mathcal{C}_1}\} \sqcup \mathcal{D}_1 \setminus \{\mathcal{L}_{\mathcal{D}_1}\} \sqcup \mathcal{L}$ with $\mathcal{L}_{\mathcal{C}_1}(,\mathcal{L}_{\mathcal{D}_1}) \Rightarrow_\alpha \mathcal{L}$ ($\mathcal{L}$ may be $\bot$). It also holds that $\mathcal{E}_2 = \mathcal{C}_2 \sqcup \mathcal{D}_2$. Now, if $\mathcal{L}_{\mathcal{C}_1} \notin \mathcal{C}^*$, then $\mathcal{C}^* \subseteq \mathcal{C}_1 \setminus \{\mathcal{L}_{\mathcal{C}_1}\} \cup \mathcal{C}_2$ and $\mathcal{C}^* \subseteq \mathcal{E}_1 \cup \mathcal{E}_2$ holds. Similarly, if $\mathcal{L}_{\mathcal{D}_1} \notin \mathcal{D}^*$, we have $\mathcal{D}^* \subseteq \mathcal{D}_1 \setminus \{\mathcal{L}_{\mathcal{D}_1}\} \cup \mathcal{D}_2$ and $\mathcal{D}^* \subseteq \mathcal{E}_1 \cup \mathcal{E}_2$. We can now assume that $\mathcal{L}_{\mathcal{C}_1} \in \mathcal{C}^*$ and $\mathcal{L}_{\mathcal{D}_1} \in \mathcal{D}^*$. It is then easy to see that $\mathcal{C}^*(,\mathcal{D}^*) \Rightarrow_\alpha \mathcal{E}^*$ with

$$\begin{aligned}\mathcal{E}^* &= \mathcal{C}^* \setminus \{\mathcal{L}_{\mathcal{C}_1}\} \sqcup \mathcal{D}^* \setminus \{\mathcal{L}_{\mathcal{D}_1}\} \sqcup \mathcal{L}\\ &\subseteq \mathcal{C}_1 \setminus \{\mathcal{L}_{\mathcal{C}_1}\} \sqcup \mathcal{D}_1 \setminus \{\mathcal{L}_{\mathcal{D}_1}\} \sqcup \mathcal{L} \sqcup \mathcal{C}_2 \sqcup \mathcal{D}_2\\ &= \mathcal{E}_1 \cup \mathcal{E}_2.\end{aligned}$$

The case for $\mathcal{C}_2(,\mathcal{D}_2) \Rightarrow_\alpha \mathcal{E}_2$ and $\mathcal{E}_1 = \mathcal{C}_1 \cup \mathcal{D}_1$ can be proved analogously. □

We now show that inferences acting on different partitions of a bipartite derivation can be reordered. In what follows, for $x \in \{l, r\}$, we set $\bar{x} = r$, whenever $x = l$, and $\bar{x} = l$, whenever $x = r$.

Lemma 33. *If $[\mathcal{C}](,[\mathcal{D}]) \Rightarrow_{[\alpha]_x} [\mathcal{I}]$ and $[\mathcal{I}](,[\mathcal{E}]) \Rightarrow_{[\beta]_{\overline{x}}} [\mathcal{F}]$, then it holds that there exists a clause $\mathcal{F}^*$ such that $[\mathcal{F}^*] \subseteq [\mathcal{F}]$ and either*

- *$[\mathcal{C}](,[\mathcal{E}]) \Rightarrow_{[\beta]_{\overline{x}}} [\mathcal{I}^*]$ and $[\mathcal{I}^*](,[\mathcal{D}]) \Rightarrow_{[\alpha]_x} [\mathcal{F}^*]$; or*
- *$[\mathcal{D}](,[\mathcal{E}]) \Rightarrow_{[\beta]_{\overline{x}}} [\mathcal{I}^*]$ and $([\mathcal{C}],)[\mathcal{I}^*] \Rightarrow_{[\alpha]_x} [\mathcal{F}^*]$; or*
- *$[\mathcal{C}], [\mathcal{E}] \Rightarrow_{[\beta]_{\overline{x}}} [\mathcal{I}_1^*]$ and $[\mathcal{D}], [\mathcal{E}] \Rightarrow_{[\beta]_{\overline{x}}} [\mathcal{I}_2^*]$ and $[\mathcal{I}_1^*], [\mathcal{I}_2^*] \Rightarrow_{[\alpha]_x} [\mathcal{F}^*]$.*

Proof. Let $\mathcal{L}_\mathcal{C} \in [\mathcal{C}]_x$ and $\mathcal{L}_\mathcal{D} \in [\mathcal{D}]_x$ such that $\mathcal{L}_\mathcal{C}(,\mathcal{L}_\mathcal{D}) \Rightarrow_\alpha \mathcal{L}_1$,

$$[\mathcal{I}]_x = [\mathcal{C}]_x \setminus \{\mathcal{L}_\mathcal{C}\} \cup [\mathcal{D}]_x \setminus \{\mathcal{L}_\mathcal{D}\} \cup \mathcal{L}_1$$

and $[\mathcal{I}]_{\overline{x}} = [\mathcal{C}]_{\overline{x}} \cup [\mathcal{D}]_{\overline{x}}$. Furthermore, let $\mathcal{L} \in [\mathcal{I}]_{\overline{x}}$, $\mathcal{L}_\mathcal{E} \in [\mathcal{E}]_{\overline{x}}$ such that $\mathcal{L}(,\mathcal{L}_\mathcal{E}) \Rightarrow_\beta \mathcal{L}_2$ and

$$\begin{aligned}[\mathcal{F}]_x &= [\mathcal{I}]_x \cup [\mathcal{E}]_x\\ &= [\mathcal{C}]_x \setminus \{\mathcal{L}_\mathcal{C}\} \cup [\mathcal{D}]_x \setminus \{\mathcal{L}_\mathcal{D}\} \cup \{\mathcal{L}_1\} \cup [\mathcal{E}]_x\end{aligned}$$

and

$$\begin{aligned}[\mathcal{F}]_{\overline{x}} &= [\mathcal{I}]_{\overline{x}} \setminus \{\mathcal{L}\} \cup [\mathcal{E}]_{\overline{x}} \setminus \{\mathcal{L}_\mathcal{E}\} \cup \{\mathcal{L}_2\}\\ &= ([\mathcal{C}]_{\overline{x}} \cup [\mathcal{D}]_{\overline{x}}) \setminus \{\mathcal{L}\} \cup [\mathcal{E}]_{\overline{x}} \setminus \{\mathcal{L}_\mathcal{E}\} \cup \{\mathcal{L}_2\}.\end{aligned}$$

Now we distinguish between the following cases for $\mathcal{L} \in [\mathcal{I}]_{\overline{x}} = [\mathcal{C}]_{\overline{x}} \cup [\mathcal{D}]_{\overline{x}}$ to hold.

If $\mathcal{L} \in [\mathcal{C}]_{\overline{x}} \cap [\mathcal{D}]_{\overline{x}}$, then there exist clauses $[\mathcal{I}_1^*]$ and $[\mathcal{I}_2^*]$ such that

$$[\mathcal{C}]_{\overline{x}}(,[\mathcal{E}]_{\overline{x}}) \Rightarrow_{[\beta]_{\overline{x}}} [\mathcal{I}_1^*]_{\overline{x}} = [\mathcal{C}]_{\overline{x}} \setminus \{\mathcal{L}\} \cup [\mathcal{E}]_{\overline{x}} \setminus \{\mathcal{L}_\mathcal{E}\} \cup \{\mathcal{L}_2\},$$

$[\mathcal{I}_1^*]_x = [\mathcal{C}]_x \cup [\mathcal{E}]_x$ and

$$[\mathcal{D}]_{\overline{x}}(,[\mathcal{E}]_{\overline{x}}) \Rightarrow_{[\beta]_{\overline{x}}} [\mathcal{I}_2^*]_{\overline{x}} = [\mathcal{D}]_{\overline{x}} \setminus \{\mathcal{L}\} \cup [\mathcal{E}]_{\overline{x}} \setminus \{\mathcal{L}_\mathcal{E}\} \cup \{\mathcal{L}_2\},$$

and $[\mathcal{I}_2^*]_x = [\mathcal{D}]_x \cup [\mathcal{E}]_x$. Additionally, we can infer that there exists a clause $[\mathcal{F}^*]$ such that

$$[\mathcal{I}_1^*]_x(,[\mathcal{I}_2^*]_x) \Rightarrow_{[\alpha]_x} [\mathcal{F}^*]_x = ([\mathcal{C}]_x \cup [\mathcal{E}]_x) \setminus \{\mathcal{L}_\mathcal{C}\} \cup ([\mathcal{D}]_x \cup [\mathcal{E}]_x) \setminus \{\mathcal{L}_\mathcal{D}\} \cup \{\mathcal{L}_1\}$$

and

$$[\mathcal{F}^*]_{\overline{x}} = [\mathcal{I}_1^*]_{\overline{x}} \cup [\mathcal{I}_2^*]_{\overline{x}} = [\mathcal{C}]_{\overline{x}} \setminus \{\mathcal{L}\} \cup [\mathcal{D}]_{\overline{x}} \setminus \{\mathcal{L}\} \cup [\mathcal{E}]_{\overline{x}} \setminus \{\mathcal{L}_\mathcal{E}\} \cup \{\mathcal{L}_2\}.$$

We can conclude that $[\mathcal{F}^*] \subseteq [\mathcal{F}]$ holds.

If $\mathcal{L} \in [\mathcal{C}]_{\overline{x}} \setminus [\mathcal{D}]_{\overline{x}}$, there exists a clause $[\mathcal{I}^*]$ such that

$$[\mathcal{C}]_{\overline{x}}(,[\mathcal{E}]_{\overline{x}}) \Rightarrow_{[\beta]_{\overline{x}}} [\mathcal{I}^*]_{\overline{x}} = [\mathcal{C}]_{\overline{x}} \setminus \{\mathcal{L}\} \cup [\mathcal{E}]_{\overline{x}} \setminus \{\mathcal{L}_{\mathcal{E}}\} \cup \{\mathcal{L}_2\}$$

and $[\mathcal{I}^*]_x = [\mathcal{C}]_x \cup [\mathcal{E}]_x$. Additionally, we can infer that there exists a clause $[\mathcal{F}^*]$ such that

$$[\mathcal{I}^*]_x(,[\mathcal{D}]_x) \Rightarrow_{[\alpha]_x} [\mathcal{F}^*]_x = ([\mathcal{C}]_x \cup [\mathcal{E}]_x) \setminus \{\mathcal{L}_{\mathcal{C}}\} \cup [\mathcal{D}]_x \setminus \{\mathcal{L}_{\mathcal{D}}\} \cup \{\mathcal{L}_1\}$$

and

$$[\mathcal{F}^*]_{\overline{x}} = [\mathcal{I}^*]_{\overline{x}} \cup [\mathcal{D}]_{\overline{x}} = [\mathcal{C}]_{\overline{x}} \setminus \{\mathcal{L}\} \cup [\mathcal{E}]_{\overline{x}} \setminus \{\mathcal{L}_{\mathcal{E}}\} \cup \{\mathcal{L}_2\} \cup [\mathcal{D}]_{\overline{x}}.$$

It is easy to see that $[\mathcal{F}^*] \subseteq [\mathcal{F}]$ holds.

The case $\mathcal{L} \in [\mathcal{D}]_{\overline{x}} \setminus [\mathcal{C}]_{\overline{x}}$ is symmetric to the case considered above. $\square$

The following example shows that the reordering of a proof can lead to the derivation of clauses that are smaller w.r.t. $\subseteq$.

Example 34. *Let $\mathcal{C}_1 = A \sqcup B, \mathcal{C}_2 = \neg A$ and $\mathcal{C}_3 = A \sqcup \neg B$. Then, $A \sqcup B, \neg A \Rightarrow_A B$ and $B, \neg B \sqcup A \Rightarrow_B A$. But $A \sqcup B, \neg B \sqcup A \Rightarrow_B A$ and $A, \neg A \Rightarrow_A \bot$.*

Definition 35. *Let Δ be a bipartite derivation. For $x \in \{l, r\}$ and an inference path $P = [\alpha_1]_{x_1}, \ldots, [\alpha_m]_{x_m}$ in Δ and resulting clauses we denote by $[P]_x$ the restriction of P on inferences performed on x only.*

For $x \in \{l, r\}$ we denote by $[\Delta]_x$ the following set of paths

$$[\Delta]_x = \{\, [P]_x \mid P \in \mathrm{InferencePath}(\Delta)\,\}$$

Definition 36. *We write $\Delta \Subset \Delta'$ if, and only if,* $\mathrm{Inferences}([\Delta]_l) \subseteq \mathrm{Inferences}([\Delta']_l)$ *and* $\mathrm{Inferences}([\Delta]_r) \subseteq \mathrm{Inferences}([\Delta']_r)$.

Definition 37 (Partial $(\mathfrak{T}, \Upsilon, \succ)$-Derivation)**.** *Let $\mathcal{N}$ be a set of bipartite clauses, and let Δ be a bipartite derivation from $\mathcal{N}$.*

We say that Δ is a partial $(\mathfrak{T}, \Upsilon, \succ)$-derivation *if, and only if,*

- *the last inference of Δ is $\mathcal{C}_1(,\mathcal{C}_2) \Rightarrow_{[\beta]_x} \mathcal{C}$ and the clauses $\mathcal{C}_1(,\mathcal{C}_2)$ have been obtained through $(\mathfrak{T}, \Upsilon, \succ)$-derivations from $\mathcal{N}$; and*
- *all paths in $[\Delta]_l$ and $[\Delta]_r$ are $(\mathfrak{T}, \Upsilon, \succ)$-paths.*

Essentially, partial $(\mathfrak{T}, \Upsilon, \succ)$ derivations are $(\mathfrak{T}, \Upsilon, \succ)$ derivations potentially 'broken' in the last derivation step. We address restructuring partial $(\mathfrak{T}, \Upsilon, \succ)$ derivations by the following lemma.

Lemma 38. *Let $[\mathcal{N}]$ be a set of bipartite clauses and let Δ be a partial $(\mathfrak{T}, \Upsilon, \succ)$-derivation of a clause $[\mathcal{F}]$ from $[\mathcal{N}]$.*

Then there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ' of a clause $[\mathcal{F}']$ such that

- *$[\mathcal{F}'] \subseteq [\mathcal{F}]$, and*
- *$\Delta' \Subset \Delta$.*

Proof. By induction on the number of inferences in Δ.

Let the last inference of Δ be $[\mathcal{E}_1](,[\mathcal{E}_2]) \Longrightarrow_{[\beta]_y} [\mathcal{F}]$. If Δ is a $(\mathfrak{T}, \Upsilon, \succ)$-derivation, there remains nothing to be shown. We can now assume that Δ is not an $(\mathfrak{T}, \Upsilon, \succ)$-derivation, which implies that β is a Υ-inference.

Let $\Delta_{[\mathcal{E}_1]}$ and $\Delta_{[\mathcal{E}_2]}$ be the $(\mathfrak{T}, \Upsilon, \succ)$-derivations of $[\mathcal{E}_1]$ and $[\mathcal{E}_2]$, respectively. It must hold that $\mathrm{Depth}(\Delta_{[\mathcal{E}_1]}) > 0$ or $\mathrm{Depth}(\Delta_{[\mathcal{E}_2]}) > 0$ as otherwise Δ would be a $(\mathfrak{T}, \succ)$-derivation. We assume w.l.o.g. that $\mathrm{Depth}(\Delta_{[\mathcal{E}_1]}) > 0$.

Now, if the derivations exist, let $[\mathcal{C}_1](,[\mathcal{C}_2]) \Longrightarrow_{[\alpha_1]_{x_1}} [\mathcal{E}_1]$ and $[\mathcal{D}_1](,[\mathcal{D}_2]) \Longrightarrow_{[\alpha_2]_{x_2}} [\mathcal{E}_2]$. We obtain the following graphical representation:

$$\begin{tikzcd}[column sep=small]
{[\mathcal{C}_1]} \arrow[dr, dash] & & {[\mathcal{C}_2]} \arrow[dl, dash, dashed] & & {[\mathcal{D}_1]} \arrow[dr, dash] & & {[\mathcal{D}_2]} \arrow[dl, dash, dashed] \\
& {[\alpha_1]_{x_1}\ [\mathcal{E}_1]} \arrow[drr, dash] & & & & {[\mathcal{E}_2]\ [\alpha_2]_{x_2}} \arrow[dll, dash, dashed] & \\
& & & {[\mathcal{F}]\ [\beta]_y} & & &
\end{tikzcd}$$

As β is an Υ-inference and as every path from $[\Delta]_y$ is an $(\mathfrak{T}, \Upsilon, \succ)$-path, it follows that every inference $i \in [\Delta]_y$ is an Υ-inference.

If $x_1 = x_2 = y$, it would follow that $x_1 \succeq y$, $x_2 \succeq y$, i.e. Δ is a $(\mathfrak{T}, \Upsilon, \succ)$-derivation. Hence, either $x_1 \neq y$ or $x_2 \neq y$ holds.

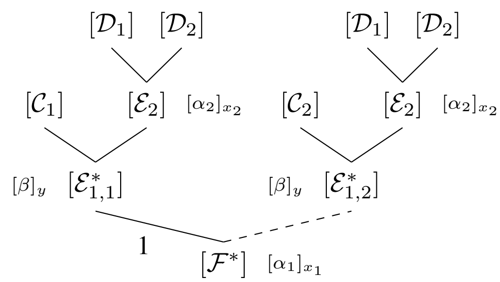
We can assume w.l.o.g. that $x_1 \neq y$ and $x_2 = y$, which implies that (α_2 is a Υ-inference and $\alpha_2 \succeq \beta$. Furthermore, we can conclude that either α_1 is not an Υ-inference, or α_1 is an Υ-inference and $\alpha_1 \not\succeq \beta$, i.e. $\beta \succ \alpha_1$, as otherwise Δ would be a $(\mathfrak{T}, \Upsilon, \succ)$-derivation.

We can apply Lemma 33 and we obtain that there exists a clause $[\mathcal{F}^*]$ such that $[\mathcal{F}^*] \subseteq [\mathcal{F}]$ and either

(i) $[\mathcal{C}_1](,[\mathcal{E}_2]) \Rightarrow_{[\beta]_y} [\mathcal{E}^*_{1,1}]$ and $[\mathcal{E}^*_{1,1}](,[\mathcal{C}_2]) \Rightarrow_{[\alpha]_{x_1}} [\mathcal{F}^*]$; or
(ii) $[\mathcal{C}_2](,[\mathcal{E}_2]) \Rightarrow_{[\beta]_y} [\mathcal{E}^*_{1,2}]$ and $([\mathcal{C}_1],)[\mathcal{E}^*_{1,2}] \Rightarrow_{[\alpha]_{x_1}} [\mathcal{F}^*]$; or
(iii) $[\mathcal{C}_1],[\mathcal{E}_2] \Rightarrow_{[\beta]_y} [\mathcal{E}^*_{1,1}]$ and $[\mathcal{C}_2],[\mathcal{E}_2] \Rightarrow_{[\beta]_y} [\mathcal{E}^*_{1,2}]$ and $[\mathcal{E}^*_{1,1}],[\mathcal{E}^*_{1,2}] \Rightarrow_{[\alpha]_{x_1}} [\mathcal{F}^*]$.

In the cases (i) and (ii) it is easy to see (due to $x_1 \neq y$ and $\beta \succ \alpha_1$) that the obtained derivation Δ' is $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ' of a clause $[\mathcal{F}^*]$ from $\mathcal{N}$ such that $[\mathcal{F}^*] \subseteq [\mathcal{F}]$ and $\Delta' \Subset \Delta$ holds.

In case (iii) we obtain the following graphical representation:



Let $\Delta_{[\mathcal{E}^*_{1,1}]}$, $\Delta_{[\mathcal{E}^*_{1,2}]}$ be the derivations of $[\mathcal{E}^*_{1,1}]$ and $[\mathcal{E}^*_{1,2}]$ as defined above.

We can infer that every y-path in $\Delta_{[\mathcal{C}_1]}$, $\Delta_{[\mathcal{C}_2]}$ and $\Delta_{[\mathcal{E}_2]}$ is an $(\mathfrak{T}, \Upsilon, \succ)$-path as every path in $[\Delta]_y$ is a $(\mathfrak{T}, \Upsilon, \succ)$-path and $x_1 \neq y$. Hence, we can conclude that every y-path in $\Delta_{[\mathcal{E}^*_{1,1}]}$ and $\Delta_{[\mathcal{E}^*_{1,2}]}$ is a $(\mathfrak{T}, \Upsilon, \succ)$-path.

Moreover, as the derivations for the clauses $[\mathcal{C}_1]$, $[\mathcal{C}_2]$, and $[\mathcal{E}_2]$ are $(\mathfrak{T}, \Upsilon, \succ)$-derivations, we can infer that every $\bar{y}$-path in $\Delta_{[\mathcal{E}^*_{1,1}]}$ and $\Delta_{[\mathcal{E}^*_{1,2}]}$ is a $(\mathfrak{T}, \Upsilon, \succ)$-path.

As the derivations $\Delta_{[\mathcal{E}^*_{1,1}]}, \Delta_{[\mathcal{E}^*_{1,2}]}$ contain less inferences than Δ, we can apply the induction hypothesis. We obtain $(\mathfrak{T}, \Upsilon, \succ)$-derivations $\Delta_{[\mathcal{E}^{**}_{1,1}]}$, $\Delta_{[\mathcal{E}^{**}_{1,2}]}$ of clauses $[\mathcal{E}^{**}_{1,1}], [\mathcal{E}^{**}_{1,2}]$ such that $[\mathcal{E}^{**}_{1,1}] \subseteq [\mathcal{E}^*_{1,1}]$, $[\mathcal{E}^{**}_{1,2}] \subseteq [\mathcal{E}^*_{1,2}]$ and $\Delta_{[\mathcal{E}^{**}_{1,1}]} \Subset \Delta_{[\mathcal{E}^*_{1,1}]}$, $\Delta_{[\mathcal{E}^{**}_{1,2}]} \Subset \Delta_{[\mathcal{E}^*_{1,2}]}$

It follows from Lemma 31 that either

- $[\mathcal{E}^{**}_{1,1}] \subseteq [\mathcal{F}^*]$, or
- $[\mathcal{E}^{**}_{1,2}] \subseteq [\mathcal{F}^*]$, or
- $[\mathcal{E}^{**}_{1,1}](,[\mathcal{E}^{**}_{1,2}]) \Longrightarrow_{\alpha_{1_{x_1}}} [\mathcal{F}^{**}]$ with $[\mathcal{F}^{**}] \subseteq [\mathcal{F}^*]$.

As α_1 is either not a Υ-inference, or α_1 is a Υ-inference and $\beta \succ \alpha_1$, which implies that $\gamma \succ \alpha_1$ for every $\gamma \in \mathrm{Inferences}(\Delta_{[\mathcal{E}^{**}_{1,1}]_{x_1}}) \cup \mathrm{Inferences}(\Delta_{[\mathcal{E}^{**}_{1,2}]_{x_1}})$, we can infer in all the cases above that there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ' of a clause $[\mathcal{F}']$ from $\mathcal{N}$ such that $[\mathcal{F}'] \subseteq [\mathcal{F}]$ and $\Delta' \Subset \Delta$ holds.

□

Next we extend the previous results by dropping the requirement that the subderivations are $(\mathfrak{T}, \Upsilon, \succ)$-derivations.

Lemma 39. *Let Δ be a bipartite $\mathfrak{T}$-derivation of a clause $[\mathcal{F}]$ from clauses in $[\mathcal{N}]$ such that all the paths in $[\Delta]_l$ and $[\Delta]_r$ are $(\mathfrak{T}, \Upsilon, \succ)$-paths.*

Then there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ^* of a clause $[\mathcal{F}^*]$ from clauses in $[\mathcal{N}]$ such that

- $[\mathcal{F}^*] \subseteq [\mathcal{F}]$*, and*
- $\Delta^* \Subset \Delta$.

Proof. By induction on the depth of Δ.

If $\mathrm{Depth}(\Delta) \leq 1$, nothing remains to be shown.

Otherwise, let the last inference of Δ be $[\mathcal{E}_1](,[\mathcal{E}_2]) \Longrightarrow_{[\beta]_y} [\mathcal{F}]$, where $\Delta_{[\mathcal{E}_1]}$ and $\Delta_{[\mathcal{E}_2]}$ are the $\mathfrak{T}$-derivations of $[\mathcal{E}_1]$ and $[\mathcal{E}_2]$, respectively.

As all the paths in $[\Delta_{\mathcal{E}_i}]_l$ and $[\Delta_{\mathcal{E}_i}]_r$ for $i \in \{1,2\}$ are $(\mathfrak{T}, \Upsilon, \succ)$-paths, it follows from the induction hypothesis that there exist $(\mathfrak{T}, \Upsilon, \succ)$-derivations $\Delta_{\mathcal{E}^*_i}$ of clauses $[\mathcal{E}^*_i]$ such that $[\mathcal{E}^*_i] \subseteq [\mathcal{E}_i]$ and $\Delta_{[\mathcal{E}^*_i]} \Subset \Delta_{[\mathcal{E}_i]}$ for $i \in \{1,2\}$.

It then follows from Lemma 31 that either

- $[\mathcal{E}^*_1] \subseteq [\mathcal{F}]$, or
- $[\mathcal{E}^*_2] \subseteq [\mathcal{F}]$, or
- $[\mathcal{E}^*_1](,[\mathcal{E}^*_2]) \Longrightarrow_{[\beta]_y} [\mathcal{F}^*]$ with $[\mathcal{F}^*] \subseteq [\mathcal{F}]$.

In the first two cases or if β is not a Υ-inference, nothing remains to be shown.

We now assume that $[\mathcal{E}_1^*](,[\mathcal{E}_2^*]) \Longrightarrow_{\beta_y} [\mathcal{F}^*]$ with $[\mathcal{F}^*] \subseteq [\mathcal{F}]$ and that β is a Υ-inference. Let $\Delta_{[\mathcal{F}^*]}$ be the derivation of the clause $\Delta_{[\mathcal{F}^*]}$.

If $\Delta_{[\mathcal{F}^*]}$ is a $(\mathfrak{T}, \Upsilon, \succ)$-derivation or β is not a Υ-inference, there remains nothing to be shown. We can now assume that $\Delta_{[\mathcal{F}^*]}$ is not a $(\mathfrak{T}, \Upsilon, \succ)$-derivation and that β is a Υ-inference.

As every y-path in Δ is a $(\mathfrak{T}, \Upsilon, \succ)$-path and as $\Delta_{[\mathcal{E}_i^*]} \Subset \Delta_{[\mathcal{E}_i]}$ for $i \in \{1,2\}$, we can conclude that every y-path in $\Delta_{[\mathcal{F}^*]}$ is a $(\mathfrak{T}, \Upsilon, \succ)$-path.

Moreover, as the derivations for the clauses $[\mathcal{E}_1^*]$ and $[\mathcal{E}_2^*]$ are $(\mathfrak{T}, \Upsilon, \succ)$-derivations, we can infer that every $\bar{y}$-path in $\Delta_{[\mathcal{E}_{1,1}^*]}$ and $\Delta_{[\mathcal{E}_{1,2}^*]}$ is a $(\mathfrak{T}, \Upsilon, \succ)$-path. We thus obtain that $\Delta_{[\mathcal{F}^*]}$ is a partial $(\mathfrak{T}, \Upsilon)$-derivation.

We can thus conclude that the statement of the Lemma holds by applying Lemma 38. ☐

Finally, we get rid of partitioning.

Lemma 40. *Let $[\mathcal{N}] = \{[\mathcal{C}_1]_l \cup [\mathcal{D}_1]_r, \ldots, [\mathcal{C}_n]_l \cup [\mathcal{D}_n]_r\}$ be a set of bipartite clauses and let Δ be a $(\mathfrak{T}, \Upsilon, \succ)$-derivation of a clause $[\mathcal{E}]$ from $[\mathcal{N}]$.*

Then there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ^* of a clause $\mathcal{E}^*$ from $\mathcal{N} = \{\mathcal{C}_1 \cup \mathcal{D}_1, \ldots, \mathcal{C}_n \cup \mathcal{D}_n\}$ such that $\mathcal{E}^* \subseteq \mathcal{E}$ and $\mathrm{Inferences}(\Delta^*) \subseteq \mathrm{Inferences}(\Delta)$.

Proof. By induction on the depth d of Δ.

If $d = 0$, then Δ simply consists of a clause $[\mathcal{C}_i]_l \cup [\mathcal{D}_i]_r$ in $[\mathcal{N}]$. The derivation Δ^* is now composed of the clause $\mathcal{C}_i \cup \mathcal{D}_i$.

For $d > 0$, let $[\mathcal{C}_1]_l \cup [\mathcal{C}_2]_r(,[\mathcal{D}_1]_l \cup [\mathcal{D}_2]_r) \Rightarrow_\alpha [\mathcal{E}] = [\mathcal{E}_1]_l \cup [\mathcal{E}_2]_r$ be the last inference of Δ, and let $\Delta_\mathcal{C}$, $\Delta_\mathcal{D}$ be the subderivations of $[\mathcal{C}_1]_l \cup [\mathcal{C}_2]_r$ and $[\mathcal{D}_1]_l \cup [\mathcal{D}_2]_r$, respectively. As $\Delta_\mathcal{C}$ and $\Delta_\mathcal{D}$ are $(\mathfrak{T}, \Upsilon, \succ)$-derivations, it follows from the induction hypothesis that there exist $(\mathfrak{T}, \Upsilon, \succ)$-derivations $\Delta^*_{\mathcal{C}^*}$ and $\Delta^*_{\mathcal{D}^*}$ of clauses $\mathcal{C}^*$ and $\mathcal{D}^*$ from $\mathcal{N}$ such that $\mathcal{C}^* \subseteq \mathcal{C}_1 \cup \mathcal{C}_2$, $\mathcal{D}^* \subseteq \mathcal{D}_1 \cup \mathcal{D}_2$, and $\mathrm{Inferences}(\Delta^*_\mathcal{C}) \subseteq \mathrm{Inferences}(\Delta_\mathcal{C})$, $\mathrm{Inferences}(\Delta^*_\mathcal{D}) \subseteq \mathrm{Inferences}(\Delta_\mathcal{D})$.

Thus, we obtain from Lemma 32 that either

(i) $\mathcal{C}^* \subseteq \mathcal{E}_1 \cup \mathcal{E}_2$, or

(ii) $\mathcal{D}^* \subseteq \mathcal{E}_1 \cup \mathcal{E}_2$, or

(iii) $\mathcal{C}^*, (\mathcal{D}^*) \Rightarrow_\alpha \mathcal{E}^*$ such that $\mathcal{E}^* \subseteq \mathcal{E}_1 \cup \mathcal{E}_2$.

In all three cases it is now easy to see that one can obtain the required derivation Δ^* of a clause $\mathcal{E}^*$ with $\mathcal{E}^* \subseteq \mathcal{E}$ and $\mathrm{Inferences}(\Delta^*) \subseteq \mathrm{Inferences}(\Delta)$. ☐

We now use bipartite derivations and Lemma 40 to prove the following result on reordering of derivation steps.

Lemma 41. *Let $\Delta_\mathcal{C}$ be a $(\mathfrak{T}, \Upsilon, \succ)$-refutation from clauses $\mathcal{M} \cup \{\mathcal{C}\}$ and let $\Delta_\mathcal{D}$ be a $(\Upsilon, \mathfrak{T}, \succ)$-refutation from clauses $\mathcal{N} \cup \{\mathcal{D}\}$.*

Then there exists a $(\mathfrak{T}, \Upsilon, \succ)$-refutation Δ from $\mathcal{M} \cup \mathcal{N} \cup \{\mathcal{C} \sqcup \mathcal{D}\}$.

Proof. If $\mathcal{C} \notin \mathrm{Leaves}(\Delta_\mathcal{C})$, then we can define $\Delta = \Delta_\mathcal{C}$. Similarly, if $\mathcal{D} \notin \mathrm{Leaves}(\Delta_\mathcal{D})$, we set $\Delta = \Delta_\mathcal{D}$. We can now assume that $\mathcal{C} \in \mathrm{Leaves}(\Delta_\mathcal{C})$ and $\mathcal{D} \in \mathrm{Leaves}(\Delta_\mathcal{C})$.

Let $[\mathcal{M}] = \{ [\mathcal{E}]_l \cup [\bot]_r \mid \mathcal{E} \in \mathcal{M} \}$ and $[\mathcal{N}] = \{ [\bot]_l \cup [\mathcal{E}]_r \mid \mathcal{E} \in \mathcal{N} \}$. It is then easy to see that there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation $[\Delta_\mathcal{C}]$ of the clause $[\bot]_l \cup [\bot]_r$ from $[\mathcal{M}] \cup \{[\mathcal{C}]_l \cup [\bot]_r\}$, and that there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation $[\Delta_\mathcal{D}]$ of the clause $[\bot]_l \cup [\bot]_r$ from $[\mathcal{N}] \cup \{[\bot]_l \cup [\mathcal{D}]_r\}$. In particular, all the paths in $[\Delta_\mathcal{D}]_r$ are $(\mathfrak{T}, \Upsilon, \succ)$-paths.

Additionally, by modifying the derivation $[\Delta_\mathcal{C}]$ it is easy to see that there exists a derivation $[\Delta']$ of the clause $[\bot]_l \cup [\mathcal{D}]_r$ from $[\mathcal{M}] \cup \{[\mathcal{C}]_l \cup [\mathcal{D}]_r\}$ such that all the paths in $[\Delta']_l$ are $(\mathfrak{T}, \Upsilon, \succ)$-paths. Then, by extending the derivation $[\Delta']$ with the derivation $[\Delta_\mathcal{D}]$, there exists a derivation Δ'' of the clause $[\bot]_l \cup [\bot]_r$ from $[\mathcal{M}] \cup [\mathcal{N}] \cup \{[\mathcal{C}]_l \cup [\mathcal{D}]_r\}$ such that all the paths in $[\Delta'']_l$ and $[\Delta'']_r$ are $(\mathfrak{T}, \Upsilon, \succ)$-paths. We can hence apply Lemma 39 and we obtain that there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ''^* of the clause $[\bot]_l \cup [\bot]_r$ from $[\mathcal{M}] \cup [\mathcal{N}] \cup \{[\mathcal{C}]_l \cup [\mathcal{D}]_r\}$.

Finally, we obtain the required $(\mathfrak{T}, \Upsilon, \succ)$-refutation Δ by applying Lemma 40. ☐

Next we focus on showing that refutations can be constructed out of closed nodes that result from type 2 expansions in a model tree. The proof proceeds by induction on the structure of clauses in the closed node. The main technical difficulty for establishing the result lies in the fact that $(\mathfrak{T}, \Upsilon, \succ)$-derivations might have to be 'linearised'.

Example 42. *Consider the following unsatisfiable clause set $\{\exists r.(A \sqcup B, \neg A, \neg B)\}$. Its refutation can be constructed inductively as follows.*

For the clause set under the existential restriction, $\{A \sqcup B, \neg A, \neg B\}$, a contradiction can be derived as follows:

$$\dfrac{\dfrac{A \sqcup B \quad \neg A}{B}\,{\scriptstyle A} \qquad \neg B}{\bot}\,{\scriptstyle B}$$

In the induction step, the 'corresponding' derivation from $\exists r.(A \sqcup B, \neg A, \neg B)$ becomes

$$\begin{array}{rc} & \exists r.(A \sqcup B, \neg A, \neg B) \\ & \mid \\ {\scriptstyle A} & \exists r.(A \sqcup B, \neg A, \neg B, B) \\ & \mid \\ {\scriptstyle B} & \bot \end{array}$$

which clearly has a different shape.

We address the necessity to linearise the proof steps by introducing a notion of an inference-preserving isomorphism as follows.

Definition 43. *For $x, y \in \{\mathfrak{T}, \mathfrak{T}^u, \mathfrak{T}^f, \mathfrak{T}^{u,f}\}$ let Δ and Δ' be x- and y-derivation trees (with labelling functions L, L') from sets of clauses $\mathcal{N}$, $\mathcal{N}'$, respectively. Let $\mathcal{M}$ be a set of initial clauses and let $\mathcal{N}$ be a set of universal clauses.*

A bijective function $f\colon \mathrm{Nodes}(\Delta) \to \mathrm{Nodes}(\Delta')$ is an inference-preserving isomorphism *if, and only if,*

- *for every $m \in \mathrm{Nodes}(\Delta)$ it holds that*

$$m \in \mathrm{Leaves}(\Delta) \quad \textit{iff} \quad f(m) \in \mathrm{Leaves}(\Delta'),$$

  and

- *for every $m, n, l \in \mathrm{Nodes}(\Delta)$ it holds that the clause $L(l)$ is obtained through a x-inference on α from the clause $L(m)$ (and potentially $L(n)$) if, and only if, the clause $L'(f(l))$ is obtained through a y-inference on α from the clause $L'(f(m))$ (and potentially $L'(f(n))$).*

We write $\Delta \rightleftharpoons \Delta'$ if, and only if, there exists an inference-preserving isomorphism $f\colon \mathrm{Nodes}(\Delta) \to \mathrm{Nodes}(\Delta')$.

Then we establish relationships between derivations from clauses contained under the universal or existential restriction and derivations at the 'higher level' required for the inductive step.

Lemma 44. *Let $x \in \{\mathfrak{T}, \mathfrak{T}^u, \mathfrak{T}^f, \mathfrak{T}^{u,f}\}$ and let Δ be a $\mathfrak{T}$-derivation Δ of a clause $\mathcal{D}$ from $\{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$.*

Then there exists a x-derivation Δ^* of a clause $\forall r.\mathcal{D}$ from $\{\forall r.\mathcal{C}_1, \ldots, \forall r.\mathcal{C}_m\}$ such that $\Delta \rightleftharpoons \Delta^*$

Proof. The proof proceeds by induction on the depth d of Δ.

If $d = 0$, it follows that $\mathcal{D} \in \{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$. Obviously, it holds that $\forall r.\mathcal{D} \in \{\forall r.\mathcal{C}_1, \ldots, \forall r.\mathcal{C}_m\}$, and the required derivation Δ^* just consists of the clause $\forall r.\mathcal{D}$.

For $d > 0$, let $\mathcal{C}'_1, \mathcal{C}'_2$ be clauses such that $\mathcal{C}'_1(, \mathcal{C}'_2) \vdash^{\alpha}_{x} \mathcal{D}$, and let $\Delta_{\mathcal{C}'_1}$ and $\Delta_{\mathcal{C}'_2}$ be the corresponding derivations of $\mathcal{C}'_1$ and $\mathcal{C}'_2$. We can also assume that $\mathcal{C}'_1(, \mathcal{C}'_2) \Rightarrow_{\alpha} \mathcal{D}$ holds. It then follows from the induction hypothesis that there exists x-derivations $\Delta^*_{\mathcal{C}'_1}$, $\Delta^*_{\mathcal{C}'_2}$ of clauses $\forall r.\mathcal{C}'_1, \forall r.\mathcal{C}'_2$, respectively, from $\{\forall r.\mathcal{C}_1, \ldots, \forall r.\mathcal{C}_m\}$ such that $\Delta_{\mathcal{C}'_1} \rightleftharpoons \Delta^*_{\mathcal{C}'_1}$ and $\Delta_{\mathcal{C}'_2} \rightleftharpoons \Delta^*_{\mathcal{C}'_2}$.

We can conclude that $\forall r.\mathcal{C}'_1 \vdash^{\alpha}_{x} \forall r.\mathcal{D}$ or $\forall r.\mathcal{C}'_1, \forall r.\mathcal{C}'_2 \vdash^{\alpha}_{x} \forall r.\mathcal{D}$ respectively holds by the $\forall$ and $\forall\forall$ rules. We have thus obtained the required derivation Δ^* with $\Delta_{\mathcal{C}'_2} \rightleftharpoons \Delta^*_{\mathcal{C}'_2}$. ☐

Corollary 45. *Let Δ be a $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ of a clause $\mathcal{D}$ from $\{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$.*

Then there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ^* of the clause $\forall r.\mathcal{D}$ from $\{\forall r.\mathcal{C}_1, \ldots, \forall r.\mathcal{C}_m\}$.

Proof. By Lemma 44 there exists a $\mathfrak{T}$-derivation Δ^* of the clause $\forall r.\mathcal{D}$ from $\{\forall r.\mathcal{C}_1, \ldots, \forall r.\mathcal{C}_m\}$ such that $\Delta \rightleftharpoons \Delta^*$. It is then easy to see that Δ^* is a $(\mathfrak{T}, \Upsilon, \succ)$-derivation. ☐

Lemma 46. *Let $x \in \{\mathfrak{T}, \mathfrak{T}^u, \mathfrak{T}^f, \mathfrak{T}^{u,f}\}$ and let Δ be a x-derivation of a clause $\mathcal{E}$ from $\{\mathcal{C}_1, \ldots, \mathcal{C}_m, \mathcal{D}_1, \ldots, \mathcal{D}_n\}$ such that $\mathrm{Leaves}(\Delta) \cap \{\mathcal{D}_1, \ldots, \mathcal{D}_n\} \neq \emptyset$, $\bot \notin \{\mathcal{D}_1, \ldots, \mathcal{D}_n\}$, and for no sub-derivation Δ' of Δ with $\mathrm{Depth}(\Delta') > 0$ it holds that $\mathrm{Leaves}(\Delta') \subseteq \{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$. Additionally, let $> \subseteq \mathrm{Nodes}(\Delta) \times \mathrm{Nodes}(\Delta)$ be a total order on $\mathrm{Nodes}(\Delta)$ such that if n_2 is a descendant of n_1, then $n_1 > n_2$ for every $n_1, n_2 \in \mathrm{Nodes}(\Delta)$.*

Then there exists an x-derivation Δ^* of a clause $\exists r.(\mathcal{E}, \mathcal{D}_1, \ldots, \mathcal{D}_n, \mathcal{F})$ from

$$\{\forall r.\mathcal{C}_1, \ldots, \forall r.\mathcal{C}_m, \exists r.(\mathcal{D}_1, \ldots, \mathcal{D}_n)\}.$$

such that

$$\begin{aligned}&\{(\text{Inference}(N_1),\ldots,\text{Inference}(N_l))\}\\&\subseteq \text{InferencePath}(\Delta^*)\\&\subseteq \{\,(\text{Inference}(N_j),\ldots,\text{Inference}(N_l)) \mid 1\leq j\leq l\,\}.\end{aligned}$$

where $> \cap (\text{Nodes}(\Delta) \setminus \text{Leaves}(\Delta))$ *is represented by the chain* $N_1 > \ldots > N_l$*.*

Proof. Let $> \cap (\text{Nodes}(\Delta) \setminus \text{Leaves}(\Delta))$ be represented by the chain $N_1 > \ldots > N_l$.

If $l = 0$, nothing remains to be shown as the derivation Δ^* then just consists of the clause $\exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n)$.

We can now assume that $l \geq 1$ and we prove for every $1 \leq i \leq l$ that there exists a derivation Δ_i^* of the clause

$$\exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n,\text{Clause}(N_1),\ldots,\text{Clause}(N_i),\mathcal{F}_i),$$

for some $\mathcal{F}_i$, from $\{\forall r.\mathcal{C}_1,\ldots,\forall r.\mathcal{C}_m,\exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n)\}$ such that

$$\begin{aligned}&\{(\text{Inference}(N_1),\ldots,\text{Inference}(N_i))\}\\&\subseteq \text{InferencePath}(\Delta^*)\\&\subseteq \{\,(\text{Inference}(N_j),\ldots,\text{Inference}(N_i) \mid 1\leq j\leq i\,\}.\end{aligned}$$

As $\text{Clause}(N_l) = \mathcal{E}$, we can then define $\Delta^* = \Delta_l^*$.

For $i = 1$, let $\mathcal{E}_1(,\mathcal{E}_2) \Rightarrow_\alpha \text{Clause}(N_1)$ with $\{\mathcal{E}_1,\mathcal{E}_2\} \subseteq \text{Leaves}(\Delta)$. It follows from the assumptions that $\{\mathcal{E}_1,\mathcal{E}_2\} \not\subseteq \{\mathcal{C}_1,\ldots,\mathcal{C}_m\}$. If $\mathcal{E}_1 \in \{\mathcal{C}_1,\ldots,\mathcal{C}_m\}$ (i.e. $\mathcal{E}_2 \in \{\mathcal{D}_1,\ldots,\mathcal{D}_n\}$), we have $\forall r.\mathcal{E}_1, \exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n) \Rightarrow_\alpha \exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n,\text{Clause}(N_1))$ by the $\forall\exists$-rule. The case for $\mathcal{E}_2 \in \{\mathcal{C}_1,\ldots,\mathcal{C}_m\}$ is similar. Finally, if $\{\mathcal{E}_1,\mathcal{E}_2\} \subseteq \{\mathcal{D}_1,\ldots,\mathcal{D}_2\}$, we obtain $\exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n) \Rightarrow_\alpha \exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n,\text{Clause}(N_1))$ by either the $\exists_1$- or $\exists_2$-rule.

For $i > 1$, it follows first from the induction hypothesis that there exists a derivation Δ_{i-1}^* of the clause

$$\exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n,\text{Clause}(N_1),\ldots,\text{Clause}(N_{i-1}),\mathcal{F}_{i-1})$$

from $\{\forall r.\mathcal{C}_1,\ldots,\forall r.\mathcal{C}_m,\exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n)\}$ such that

$$\begin{aligned}&\{(\text{Inference}(N_1),\ldots,\text{Inference}(N_{i-1}))\}\\&\subseteq \text{InferencePath}(\Delta^*)\\&\subseteq \{\,(\text{Inference}(N_j),\ldots,\text{Inference}(N_{i-1}) \mid 1\leq j\leq i-1\,\}.\end{aligned}$$

Let $\mathcal{E}_1(,\mathcal{E}_2) \Rightarrow_\alpha \text{Clause}(N_i)$ with $N_{\mathcal{E}_1}, N_{\mathcal{E}_2} \in \text{Nodes}(\Delta)$ such that $\text{Clause}(N_{\mathcal{E}_1}) = \mathcal{E}_1$ and $\text{Clause}(N_{\mathcal{E}_2}) = \mathcal{E}_2$, i.e. N_i is a descendant of $N_{\mathcal{E}_1}$ (and $N_{\mathcal{E}_1}$) in Δ. It then follows from the assumptions that $\{\mathcal{E}_1,\mathcal{E}_2\} \subseteq \text{Leaves}(\Delta) \cup \{\text{Clause}(N_1),\ldots,\text{Clause}(N_{i-1})\}$. Moreover, it follows from the assumptions that $\{\mathcal{E}_1,\mathcal{E}_2\} \not\subseteq \{\mathcal{C}_1,\ldots,\mathcal{C}_m\}$.

If $\mathcal{E}_1 \in \{\mathcal{C}_1,\ldots,\mathcal{C}_m\}$ (i.e. $\mathcal{E}_2 \in \{\mathcal{D}_1,\ldots,\mathcal{D}_n\} \cup \{\text{Clause}(N_1),\ldots,\text{Clause}(N_{i-1})\}$), we have

$$\begin{aligned}&\forall r.\mathcal{E}_1, \exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n,\text{Clause}(N_1),\ldots,\text{Clause}(N_{i-1}),\mathcal{F}_{i-1})\\&\qquad\Rightarrow_\alpha \exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n,\text{Clause}(N_1),\ldots,\text{Clause}(N_i),\mathcal{F}_{i-1})\end{aligned}$$

by the $\forall\exists$-rule. The case for $\mathcal{E}_2 \in \{\mathcal{C}_1,\ldots,\mathcal{C}_m\}$ is similar.

Finally, if $\{\mathcal{E}_1,\mathcal{E}_2\} \cap \{\mathcal{C}_1,\ldots,\mathcal{C}_n\} = \emptyset$, we obtain

$$\begin{aligned}&\exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n,\text{Clause}(N_1),\ldots,\text{Clause}(N_{i-1}),\mathcal{F}_{i-1})\\&\qquad\Rightarrow_\alpha \exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n,\text{Clause}(N_1),\ldots,\text{Clause}(N_i),\mathcal{F}_{i-1})\end{aligned}$$

by either the $\exists_1$- or $\exists_2$-rule. We have thus constructed the derivation Δ_i^* with the required properties. □

Corollary 47. *Let* Δ *be a* $(\mathfrak{T},\Upsilon,\succ)$*-derivation of a clause* $\mathcal{E}$ *from* $\{\mathcal{C}_1,\ldots,\mathcal{C}_m,\mathcal{D}_1,\ldots,\mathcal{D}_n\}$ *such that* $\text{Leaves}(\Delta) \cap \{\mathcal{D}_1,\ldots,\mathcal{D}_n\} \neq \emptyset$ *and* $\perp \notin \{\mathcal{D}_1,\ldots,\mathcal{D}_n\}$*.*

Then there exists a $(\mathfrak{T},\Upsilon,\succ)$*-derivation* Δ^* *of a clause* $\exists r.(\mathcal{E},\mathcal{D}_1,\ldots,\mathcal{D}_n,\mathcal{F})$ *from*

$$\{\forall r.\mathcal{C}_1,\ldots,\forall r.\mathcal{C}_m,\exists r.(\mathcal{D}_1,\ldots,\mathcal{D}_n)\}.$$

Proof. Let $\Delta_1^\forall,\ldots,\Delta_p^\forall$ be all the maximal subderivations of Δ such that $\text{Leaves}(\Delta_i^\forall) \subseteq \{\mathcal{C}_1,\ldots,\mathcal{C}_m\}$ for every $1 \leq i \leq m$, and let $\mathcal{C}_i^\forall$ be the clause derived in each derivation $\Delta_i^\forall$ for $1 \leq i \leq p$. Additionally, let Δ' be the subderivation of Δ that consists of all the inferences that are contained in Δ but not in $\Delta_i^\forall$ for $1 \leq i \leq p$. Hence, we have $\text{Leaves}(\Delta') \cap \{\mathcal{D}_1,\ldots,\mathcal{D}_n\} \neq \emptyset$, $\text{Leaves}(\Delta') \subseteq \{\mathcal{D}_1,\ldots,\mathcal{D}_n\} \cup \{\mathcal{C}_1^\forall,\ldots,\mathcal{C}_p^\forall\}$ and for no sub-derivation Δ'' of Δ' with $\text{Depth}(\Delta'') > 0$ it holds that $\text{Leaves}(\Delta'') \subseteq \{\mathcal{C}_1^\forall,\ldots,\mathcal{C}_p^\forall\}$.

As all the inference paths in Δ' are $(\mathfrak{T}, \Upsilon, \succ)$-paths, it is easy to see that one can construct a relation $> \subseteq \mathrm{Nodes}(\Delta') \times \mathrm{Nodes}(\Delta')$ such that $\mathrm{Inference}(N_1) \succeq \ldots \succeq \mathrm{Inference}(N_q)$ for $\{ N \in \mathrm{Nodes}(\Delta') \setminus \mathrm{Leaves}(\Delta') \mid \mathrm{Inference}(N) \subseteq \Upsilon \} = \{N_1, \ldots, N_q\}$ and if n_2 is a descendant of n_1, then $n_1 > n_2$ for every $n_1, n_2 \in \mathrm{Nodes}(\Delta')$.

Then, by applying Lemma 44 we obtain derivations $\Delta_i^{\forall,*}$ such that $\Delta_i^{\forall} \rightleftharpoons \Delta_i^{\forall,*}$ for every $1 \leq i \leq p$. Moreover, by combining the derivations $\Delta_i^{\forall,*}$ $(1 \leq i \leq p)$ with the derivation Δ'^{*} obtained from Lemma 46 applied on the derivation Δ' using the relation $>$, we can construct the required $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ^*. □

Corollary 48. *Let Δ be a $(\mathfrak{T}, \Upsilon, \succ)$-refutation from $\{\mathcal{C}_1, \ldots, \mathcal{C}_m, \mathcal{D}_1, \ldots, \mathcal{D}_n\}$ with $n \geq 1$.*
Then there exists a $(\mathfrak{T}, \Upsilon, \succ)$-refutation Δ^* from $\{\forall r.\mathcal{C}_1, \ldots, \forall r.\mathcal{C}_m, \exists r.(\mathcal{D}_1, \ldots, \mathcal{D}_n)\}$.

Proof. If $\mathrm{Leaves}(\Delta) \subseteq \{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$, it follows from Lemma 45 that there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ' of the clause $\forall r.\bot$ from $\{\forall r.\mathcal{C}_1, \ldots, \forall r.\mathcal{C}_m\}$. Consequently, we can the extend the derivation Δ' by applying the $\bot$ rule

$$\forall r.\bot, \exists r.(\mathcal{D}_1, \ldots, \mathcal{D}_n) \Longrightarrow_\bot \bot$$

We have thus constructed the required $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ^*.

Otherwise, we have $\mathrm{Leaves}(\Delta) \cap \{\mathcal{D}_1, \ldots, \mathcal{D}_n\} \neq \emptyset$. It then follows from Corollary 47 that there exists a $(\mathfrak{T}, \Upsilon)$-derivation of the clause $\exists r.(\bot, \mathcal{D}_1, \ldots, \mathcal{D}_n, \mathcal{F})$ from $\{\forall r.\mathcal{C}_1, \ldots, \forall r.\mathcal{C}_m, \exists r.(\mathcal{D}_1, \ldots, \mathcal{D}_n)\}$.

Finally, it suffices to observe that $\exists r.(\bot, D_1, \ldots, D_n, \mathcal{F})$ simplifies to $\bot$. □

Lemma 49. *Let $\mathcal{G} = \mathcal{M}$ be a model tree for $\mathcal{N}$ w.r.t. $\mathcal{T}$ and let n be a closed node in $\mathcal{G}$.*
Then there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation of the empty clause from clauses in n.

Proof. Let T be the closed tree with root n that is obtained from Lemma 27. The proof now proceeds by induction on the depth d of T.

For $d = 0$, it follows that n' contains a pair $A, \neg A$ for a concept name A. Hence, there exists a derivation of the empty clause from n', namely $A, \neg A \Longrightarrow_A \bot$.

If $d > 0$ and n is of type 2, let

$$L(n) = \{\mathcal{L}_1, \ldots, \mathcal{L}_m\} \cup \bigcup_{i=1}^{l} \{\exists r_i.\mathcal{F}_i\} \cup \bigcup_{j=1}^{p} \{\forall s_j.\mathcal{C}_j\}$$

Additionally, let n' be its closed r-child node in T and let T' be the subtree of T with root n'. Then, there exists $1 \leq k \leq l$ such that

$$L(n') = \mathcal{F}_k \cup \{ \mathcal{C}_j \mid 1 \leq j \leq p \text{ and } r_k = s_j \}.$$

It follows from the induction hypothesis that there exists a $(\mathfrak{T}, \Upsilon, \succ)$-refutation of n'. We can thus conclude from Corollary 48 that there exists a $(\mathfrak{T}, \Upsilon, \succ)$-refutation of n.

Finally, in the case where $d > 0$ such that n is of type 1, let $\mathcal{C} = \mathcal{C}_1 \sqcup \mathcal{C}_2$ be the clause that is split. Then, the children of the node n' are $n'_1 = n \setminus \{\mathcal{C}\} \cup \{\mathcal{C}_1\}$, and $n'_2 = n \setminus \{\mathcal{C}\} \cup \{\mathcal{C}_2\}$. It follows from the induction hypothesis that there exists a $(\mathfrak{T}, \Upsilon, \succ)$-refutation Δ_1 of n'_1 and a $(\mathfrak{T}, \Upsilon, \succ)$-refutation Δ_2 of n'_2. We can then apply Lemma 41 and we obtain the required $(\mathfrak{T}, \Upsilon, \succ)$-refutation Δ of n. □

The following lemma follows from the definition of the rules of $\Rightarrow_\alpha$ in a straightforward way.

Lemma 50. *Let $\mathcal{I}$ be an interpretation with domain $\Delta^{\mathcal{I}}$. Additionally, let $\mathcal{C}$ and $\mathcal{D}$ be two clauses and let $a \in \Delta^{\mathcal{I}}$ such that $a \in \mathcal{C}^{\mathcal{I}} \cap \mathcal{D}^{\mathcal{I}}$ and $\mathcal{C}(, \mathcal{D}) \Rightarrow_\alpha \mathcal{E}$.*
Then it holds that $a \in \mathcal{E}^{\mathcal{I}}$.

We are now in the position to prove Theorem 6.

Theorem 6 ($\mathfrak{T}$-Completeness)**.** *Let $\Upsilon \subseteq \mathsf{N_C}$, let $\succ \subseteq \Upsilon \times \Upsilon$ be a strict total order on Υ and let C and D be $\mathcal{ALC}$ concepts. Then it holds that $\models C \sqsubseteq D$ iff there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation of the empty clause from the initial clauses* $\mathrm{Cls}(C \sqcap \neg D)$.

Proof. First we assume that $\models C \sqsubseteq D$, which is equivalent to $\models C \sqcap \neg D \sqsubseteq \bot$. If we now assume towards a contradiction that the model tree $\mathcal{M}$ for $\mathrm{Cls}(C \sqcap \neg D)$ is open, it would follow from Lemma 25 that there exists an interpretation $\mathcal{I}$ with $\bigcap_{\mathcal{E} \in \mathrm{Cls}(C \sqcap \neg D)} \mathcal{E}^{\mathcal{I}} \not\subseteq \bot^{\mathcal{I}}$, which is equivalent to

$$\begin{aligned} \top^{\mathcal{I}} \not\subseteq \left(\neg \bigsqcap_{\mathcal{E} \in \mathrm{Cls}(C \sqcap \neg D)} \mathcal{E}\right)^{\mathcal{I}} &= (\neg(C \sqcap \neg D))^{\mathcal{I}} \\ &= (\neg C \sqcup D)^{\mathcal{I}}, \end{aligned}$$

which contradicts our assumptions. Thus, the model tree $\mathcal{M}$ with root n_0 is closed and by Lemma 49 there hence exists a $(\mathfrak{T}, \Upsilon, \succ)$-refutation Δ from $\mathrm{Cls}(C \sqcap \neg D)$.

For the inverse direction, we assume that there exists a $(\mathfrak{T}, \Upsilon, \succ)$-derivation Δ of the empty clause from the set of initial clauses $\mathrm{Cls}(C \sqcap \neg D)$. Let $\mathcal{I}$ be an interpretation with domain $\Delta^{\mathcal{I}}$. If we assume towards a contradiction that $(C \sqcap \neg D)^{\mathcal{I}} \neq \emptyset$, then as $(C \sqcap \neg D)^{\mathcal{I}} = \bigcap_{\mathcal{E} \in \mathrm{Cls}(C \sqcap \neg D)} \mathcal{E}^{\mathcal{I}}$, let $a \in \Delta^{\mathcal{I}}$ such that $a \in \mathcal{E}^{\mathcal{I}}$ for every $\mathcal{E} \in \mathrm{Cls}(C \sqcap \neg D)$. By induction on the structure of Δ and by using Lemma 50 one can show that $a \in \bot^{\mathcal{I}} = \emptyset$ would hold, which is obviously a contradiction. Thus, we can infer that $(C \sqcap \neg D)^{\mathcal{I}} = \emptyset$, which implies that $\top^{\mathcal{I}} \subseteq (\neg C \sqcup D)^{\mathcal{I}}$. □

C Correspondence Between $\mathfrak{T}$ and $\mathfrak{T}^u$

In this section we prove refutational completeness of $\mathfrak{T}^u$ with ordering constraints by reduction to the results of the previous section.

Definition 51. *Let $\mathbf{r_1}, \mathbf{r_2} \in \mathcal{R}$. We write $\mathbf{r_1} < \mathbf{r_2}$ if, and only if, there exists $\epsilon \neq \mathbf{r} \in \mathcal{R}$ such that $\mathbf{r_2} = \mathbf{r_1}.\mathbf{r}$, where . denotes concatenation. The reflexive closure of the relation $<$ on $\mathcal{R}$ is denoted by $\leq$.*

Definition 52. *Let $\mathcal{S} \subseteq \mathcal{R}$. We say that*

- *$\mathbf{r} \in \mathcal{R}$ is a common subsequence of $\mathcal{S}$ if, and only if, $\mathbf{r} \leq \mathbf{s}$ for every $\mathbf{s} \in \mathcal{S}$, and*
- *$\mathbf{r} \in \mathcal{R}$ is a greatest common subsequence of $\mathcal{S}$ if, and only if, $\mathbf{r}$ is a common subsequence of $\mathcal{S}$ and for every common subsequence $\mathbf{s}$ of $\mathcal{S}$ it holds that $\mathbf{s} \leq \mathbf{r}$.*

It is easy to see that the greatest common subsequence of a set $\mathcal{S} \subseteq \mathcal{R}$ always exists and it is unique. In the following it will be denoted by $\mathrm{gcs}(\mathcal{S})$.

Lemma 53. *Let $\mathcal{S}_1, \mathcal{S}_2 \subseteq \mathcal{R}$ and let $\mathbf{r_1} = \mathrm{gcs}(\mathcal{S}_1)$, $\mathbf{r_2} = \mathrm{gcs}(\mathcal{S}_2)$. Then*

$$\mathrm{gcs}(\mathcal{S}_1 \cup \mathcal{S}_2) = \mathrm{gcs}\{\mathbf{r_1}, \mathbf{r_2}\}$$

Proof. Let $\mathbf{r} = \mathrm{gcs}(\mathcal{S}_1 \cup \mathcal{S}_2)$. Hence, we have $\mathbf{r} \leq \mathbf{s}$ for every $\mathbf{s} \in \mathcal{S}_1$ and $\mathbf{r} \leq \mathbf{s}$ for every $\mathbf{s} \in \mathcal{S}_2$, which implies that $\mathbf{r} \leq \mathbf{r_1}$ and $\mathbf{r} \leq \mathbf{r_2}$, i.e. $\mathbf{r}$ is a common subsequence of $\{\mathbf{r_1}, \mathbf{r_2}\}$.

Let now $\mathbf{s}$ be a common subsequence of $\{\mathbf{r_1}, \mathbf{r_2}\}$, i.e. $\mathbf{s} \leq \mathbf{r_1}$ and $\mathbf{s} \leq \mathbf{r_2}$. Thus, for any $\mathbf{t} \in \mathcal{S}_1$ it holds that $\mathbf{s} \leq \mathbf{t}$ and similarly, for any $\mathbf{t} \in \mathcal{S}_2$ we have $\mathbf{s} \leq \mathbf{t}$. We can infer that $\mathbf{s} \leq \mathbf{t}$ holds for every $\mathbf{t} \in \mathcal{S}_1 \cup \mathcal{S}_2$, which implies that $\mathbf{s} \leq \mathbf{r}$.

We have thus established that $\mathbf{r}$ is a greatest common subsequence of $\{\mathbf{r_1}, \mathbf{r_2}\}$, i.e. $\mathrm{gcs}(\mathcal{S}_1 \cup \mathcal{S}_2) = \mathrm{gcs}\{\mathbf{r_1}, \mathbf{r_2}\}$. □

Lemma 54. *Let $x \in \{\mathfrak{T}, \mathfrak{T}^f\}$ and let $\mathcal{C}, \forall \mathbf{r}.\mathcal{D} \vdash_x^\alpha \mathcal{E}$ where $\mathcal{C}$ is an initial clause and $\mathcal{D}$ is a universal clause. Then there exists an inference $\mathcal{C}, \mathcal{D} \vdash_{x^u}^\alpha \mathcal{E}$.*

Proof. Follows from an application of the (mix) rule. □

Lemma 55. *Let $x \in \{\mathfrak{T}, \mathfrak{T}^f\}$ and let $\forall \mathbf{r}.\mathcal{C} \vdash_x^\alpha \mathcal{E}$ where $\mathcal{C}$ is a universal clause. Then, there exists an inference $\mathcal{C} \vdash_{x^u}^\alpha \mathcal{F}$ such that $\mathcal{F}$ is a universal clause and $\mathcal{E} = \forall \mathbf{r}.\mathcal{F}$.*

Proof. It follows from the definition of the relation $\Rightarrow_\alpha^{(f)}$ that $\mathcal{C} \Rightarrow_\alpha^{(f)} \mathcal{F}$ with $\mathcal{E} = \forall \mathbf{r}.\mathcal{F}$. We can thus conclude that $\mathcal{C} \vdash_{x^u} \mathcal{F}$ holds by applying the (u_1)-rule. □

Lemma 56. *Let $x \in \{\mathfrak{T}, \mathfrak{T}^f\}$ and let $\forall \mathbf{r_1}.\mathcal{C}, \forall \mathbf{r_2}.\mathcal{D} \vdash_x^\alpha \mathcal{E}$ where $\mathcal{C}, \mathcal{D}$ are universal clauses.*

Then, for $\mathbf{r} = \mathrm{gcs}\{\mathbf{r_1}, \mathbf{r_2}\}$, there exists an inference $\mathcal{C}, \mathcal{D} \vdash_{x^u}^\alpha \mathcal{F}$ such that $\mathcal{F}$ is a universal clause and $\mathcal{E} = \forall \mathbf{r}.\mathcal{F}$.

Proof. If we assume towards a contradiction that $\mathbf{r} \notin \{\mathbf{r_1}, \mathbf{r_2}\}$, it would hold that $\mathbf{r} < \mathbf{r_1}$ and $\mathbf{r} < \mathbf{r_2}$. Let $\mathbf{r_1} = \mathbf{r}.\mathbf{s_1}$ and $\mathbf{r_2} = \mathbf{r}.\mathbf{s_2}$ with $\mathbf{s_1}, \mathbf{s_2} \neq \epsilon$. We could infer that $(\mathbf{s_1})_0 \neq (\mathbf{s_2})_0$, where $(\mathbf{s_i})_0$ denotes $\mathbf{s_{i,0}}$ if $\mathbf{s_i} = (s_{i,0}, \ldots, s_{i,n})$ for $i \in \{1, 2\}$. It is then easy to see that $\forall \mathbf{r_1}.\mathcal{C}, \forall \mathbf{r_2}.\mathcal{D} \not\vdash_{\mathfrak{T}}^\alpha \mathcal{E}$.

Hence, we have $\mathbf{r} = \mathbf{r_1}$ or $\mathbf{r} = \mathbf{r_2}$. If $\mathbf{r} = \mathbf{r_1}$, let $\mathbf{r_2} = \mathbf{r}.\mathbf{s}$ with $\mathbf{s} \in \mathcal{R}$. It follows from the definition of the relation $\Longrightarrow_\alpha^{(f)}$ that $\mathcal{C}, \forall \mathbf{s}.\mathcal{D} \Longrightarrow_\alpha^{(f)} \mathcal{F}$ and $\mathcal{E} = \forall \mathbf{r}.\mathcal{F}$. It remains to observe that by applying the (u_2)- (if $\mathbf{s} = \epsilon$) or the (mix)-rule we obtain $\mathcal{C}, \mathcal{D} \vdash_{x^u}^\alpha \mathcal{F}$.

The case for $\mathbf{r} = \mathbf{r_2}$ can be proved similarly. □

Definition 57. *For $x, y \in \{\mathfrak{T}, \mathfrak{T}^f, \mathfrak{T}^u, \mathfrak{T}^{u,f}\}$ let Δ and Δ' be x- and y-derivation trees (with labelling functions $\mathcal{L}$, $\mathcal{L}'$) from sets of clauses $\mathcal{N}$, $\mathcal{N}'$, respectively. Let $\mathcal{M}$ be a set of initial clauses and let $\mathcal{N}$ be a set of universal clauses.*

A bijective function $f\colon \mathrm{Nodes}(\Delta) \to \mathrm{Nodes}(\Delta')$ is an inference- & depth-preserving isomorphism *if, and only if, f is an inference-preserving isomorphism and for every $n \in \mathrm{Nodes}(\Delta)$ it holds that $\mathrm{Depth}(n) \geq \mathrm{Depth}(f(n))$.*

We write $\Delta \rightleftharpoons^d \Delta^u$ if, and only if, there exists an inference & depth-preserving isomorphism $f\colon \mathrm{Nodes}(\Delta) \to \mathrm{Nodes}(\Delta^u)$.

Lemma 58. *Let $\mathcal{M}, \mathcal{M}'$ be sets of initial clauses and let $\mathcal{N}$ be a set of universal clauses. Additionally, let Δ be a $(\mathfrak{T}, \succ, \Upsilon)$-derivation of a clause $\mathcal{C}$ from $\mathcal{M}$ such that there exists $n \in \mathbb{N}$ with $\mathrm{Depth}(\mathcal{D}) \leq n$ for every $\mathcal{D} \in$ Clauses(Δ). Finally, let Δ^u be a $\mathfrak{T}^u$-derivation of a clause $\mathcal{D}^u$ from $\mathcal{M}' \cup \mathcal{N}$ such that $\Delta \rightleftharpoons^d \Delta^u$.*

Then it holds that Δ^u is a $(\mathfrak{T}^u, \succ, \Upsilon)$-derivation and $\mathrm{Depth}(\mathcal{D}^u) \leq n$ for every $\mathcal{D}^u \in$ Clauses(Δ).

Proof. By induction on the depth d of Δ using the properties of $\rightleftharpoons^d$. $\square$

Lemma 59. *Let $x \in \{\mathfrak{T}, \mathfrak{T}^f\}$, let $\mathcal{N}$ be a set of universal clauses, and let Δ be a x-derivation of a clause $\mathcal{C}$ from* $\mathrm{Univ}(\mathcal{N})$.
Then there exists a x^u-derivation Δ^u of a universal clause $\mathcal{C}'$ from $\mathcal{N}$ such that

- $\mathcal{C} = \forall \mathbf{r}.\mathcal{C}'$ *where* $\mathbf{r} = \mathrm{gcs}\{\, \mathbf{s} \mid \forall \mathbf{s}.\mathcal{D} \in \mathrm{Leaves}(\Delta), \mathcal{D} \in \mathcal{N} \,\}$*; and*
- $\Delta \rightleftharpoons^d \Delta^u$.

Proof. By induction on the depth d of Δ.

If $d = 0$, we have $\mathrm{Clauses}(\Delta) \subseteq \mathrm{Leaves}(\Delta) = \{\mathcal{C}\} = \{\forall \mathbf{r}.\mathcal{D}\}$ with $\mathcal{D} \in \mathcal{N}$. Thus, by setting $\mathcal{C}' = \mathcal{D}$ and by defining the derivation Δ^u to simply consist of the clause $\mathcal{C}'$, we obtain $\Delta \rightleftharpoons^d \Delta^u$.

In the case where $d > 0$, we distinguish between the following cases. If $\mathcal{C}$ was obtained through an application of the (i_2)-rule, let $\mathcal{C}_1$ and $\mathcal{C}_2$ be the premises used in the rule application and let $\Delta_{\mathcal{C}_1}, \Delta_{\mathcal{C}_2}$ be the respective (sub)derivations of $\mathcal{C}_1$, $\mathcal{C}_2$. It follows from the induction hypothesis that there exist x^u-derivations $\Delta^u_{\mathcal{C}_1}$ and $\Delta^u_{\mathcal{C}_2}$ of universal clauses $\mathcal{C}'_1$ and $\mathcal{C}'_2$ from $\mathcal{N}$, respectively, such that

- $\mathcal{C}_1 = \forall \mathbf{r_1}.\mathcal{C}'_1$ where $\mathbf{r_1} = \mathrm{gcs}\{\, \mathbf{s} \mid \forall \mathbf{s}.\mathcal{D} \in \mathrm{Leaves}(\Delta_{\mathcal{C}_1}), \mathcal{D} \in \mathcal{N} \,\}$, and
- $\Delta_{\mathcal{C}_1} \rightleftharpoons^d \Delta^u_{\mathcal{C}_1}$, and
- $\mathcal{C}_2 = \forall \mathbf{r_2}.\mathcal{C}'_2$ where $\mathbf{r_2} = \mathrm{gcs}\{\, \mathbf{s} \mid \forall \mathbf{s}.\mathcal{D} \in \mathrm{Leaves}(\Delta_{\mathcal{C}_2}), \mathcal{D} \in \mathcal{N} \,\}$, and
- $\Delta_{\mathcal{C}_2} \rightleftharpoons^d \Delta^u_{\mathcal{C}_2}$.

By Lemma 56, for $\mathbf{r} = \mathrm{gcs}\{\mathbf{r_1}, \mathbf{r_2}\}$ there exists an inference $\mathcal{C}_1, \mathcal{C}_2 \vdash^{\alpha}_{x^u} \mathcal{C}'$ such that $\mathcal{C} = \forall \mathbf{r}.\mathcal{C}'$. It follows from Lemma 53 that

$$\mathbf{r} = \mathrm{gcs}\{\, \mathbf{s} \mid \forall \mathbf{s}.\mathcal{D} \in \mathrm{Leaves}(\Delta), \mathcal{D} \in \mathcal{N} \,\}.$$

And by using the inference above to obtain the required derivation Δ^u, it is easy to see that $\Delta \rightleftharpoons \Delta^u$ holds.

Finally, the case where the clause $\mathcal{C}$ was obtained through an application of the (i_1)-rule can be handled similarly by applying Lemma 55. $\square$

Lemma 60. *Let $x \in \{\mathfrak{T}, \mathfrak{T}^f\}$, let $\mathcal{M}$ be a set of initial clauses, and let $\mathcal{N}$ be a set of universal clauses. Additionally, let Δ be a x-derivation of a clause $\mathcal{C}$ from $\mathcal{M} \cup \mathrm{Univ}(N)$ such that $\mathrm{Leaves}(\Delta) \cap \mathcal{M} \neq \emptyset$.*
Then there exists a x^u-derivation Δ^u of the initial clause $\mathcal{C}$ from $\mathcal{M} \cup \mathcal{N}$ such that $\Delta \rightleftharpoons^d \Delta^u$.

Proof. By induction on the depth d of Δ.

If $d = 0$, we have $\mathrm{Clauses}(\Delta) \subseteq \mathrm{Leaves}(\Delta) = \{\mathcal{C}\} \subseteq \mathcal{M}$. Thus, by defining the derivation Δ^u to simply consist of the clause $\mathcal{C}$, we obtain $\Delta \rightleftharpoons^d \Delta^u$.

In the case where $d > 0$, we distinguish between the following cases. If $\mathcal{C}$ was obtained through an application of the (i_2)-rule, let $\mathcal{C}_1$ and $\mathcal{C}_2$ be the premises used in the rule application and let $\Delta_{\mathcal{C}_1}, \Delta_{\mathcal{C}_2}$ be the respective (sub)derivations of $\mathcal{C}_1$ and $\mathcal{C}_2$.

It follows from the assumptions that either $\mathrm{Leaves}(\Delta_{\mathcal{C}}) \cap \mathcal{M} \neq \emptyset$ or $\mathrm{Leaves}(\Delta_{\mathcal{D}}) \cap \mathcal{M} \neq \emptyset$ as otherwise $\mathrm{Leaves}(\Delta) \cap \mathcal{M} = \emptyset$.

Now, if $\mathrm{Leaves}(\Delta_{\mathcal{C}}) \cap \mathcal{M} \neq \emptyset$ but $\mathrm{Leaves}(\Delta_{\mathcal{D}}) \cap \mathcal{M} = \emptyset$, we first obtain from the induction hypothesis that there exists a x^u-derivation $\Delta^u_{\mathcal{C}_1}$ of the initial clause $\mathcal{C}_1$ from $\mathcal{M} \cup \mathcal{N}$ such that $\Delta_{\mathcal{C}_1} \rightleftharpoons^d \Delta^u_{\mathcal{C}_1}$. Additionally, it follows from Lemma 59 that there exists a x^u-derivation $\Delta^u_{\mathcal{C}'_2}$ of a universal clause $\mathcal{C}'_2$ from $\mathcal{N}$ such that

- $\mathcal{C}_2 = \forall \mathbf{r}.\mathcal{C}'_2$ where $\mathbf{r} = \mathrm{gcs}\{\, \mathbf{s} \mid \forall \mathbf{s}.\mathcal{D} \in \mathrm{Leaves}(\Delta_{\mathcal{C}_2}), \mathcal{D} \in \mathcal{N} \,\}$; and
- $\Delta_{\mathcal{C}_2} \rightleftharpoons^d \Delta^u_{\mathcal{C}_2}$.

Hence, we can apply Lemma 54 and we obtain an inference $\mathcal{C}_1, \mathcal{C}'_2 \vdash^{\alpha}_{x^u} \mathcal{C}$. By combining the derivations $\Delta^u_{\mathcal{C}_1}$ and $\Delta^u_{\mathcal{C}'_2}$ together with the inference above, it is hence easy to see that one obtains the required derivation Δ^u of $\mathcal{C}$ such that $\Delta \rightleftharpoons^d \Delta^u$.

The case of $\mathrm{Leaves}(\Delta_{\mathcal{C}}) \cap \mathcal{M} = \emptyset$ but $\mathrm{Leaves}(\Delta_{\mathcal{D}}) \cap \mathcal{M} \neq \emptyset$ can be handled analogously. And if $\mathrm{Leaves}(\Delta_{\mathcal{C}}) \cap \mathcal{M} \neq \emptyset$ and $\mathrm{Leaves}(\Delta_{\mathcal{D}}) \cap \mathcal{M} \neq \emptyset$, the required derivation Δ^u can be constructed by applying the induction hypothesis twice.

Finally, the case where $\mathcal{C}$ has been derived through an application of the (i_1) rule can be proved similarly. $\square$

Theorem 7 now follows from Lemmata 58 and 60.

D Proofs for Establishing the Correctness of Algorithm 1

Before we can continue with establishing the correctness of the uniform interpolation algorithm, we have to prove a variant of the subsumption lemma for the following so-called *minimal subsumption relation* $\leq_s$ on $\mathcal{ALC}$-clauses (Auffray, Enjalbert, and Hébrard 1990), which is defined below.

Definition 61. *We define a relation $\leq_s \subseteq$ Clauses $\times$ Clauses on clauses inductively as follows:*

- *for every concept name $A \in \mathsf{N_C}$, $A \leq_s A$ and $\neg A \leq_s \neg A$,*
- *for two literals $\forall r.\mathcal{C}$, $\forall r.\mathcal{D}$, $\forall r.\mathcal{C} \leq_s \forall r.\mathcal{D}$ if, and only if, $\mathcal{C} \leq_s \mathcal{D}$,*
- *for two literals $\exists r.\mathcal{E}_1$, $\exists r.\mathcal{E}_2$, $\exists r.\mathcal{E}_1 \leq_s \exists r.\mathcal{E}_2$ if, and only if, for every clause $\mathcal{C}' \in \mathcal{E}_2$ there exists a clause $\mathcal{C} \in \mathcal{E}_1$ such that $\mathcal{C} \leq_s \mathcal{C}'$,*
- *for two initial clauses or two universal clauses $\mathcal{C}$, $\mathcal{D}$, $\mathcal{C} \leq_s, \mathcal{D}$ if, and only if, for every literal $\mathcal{L} \in \mathcal{C}$ there exists a literal $\mathcal{L}' \in \mathcal{D}$ such that $\mathcal{L} \leq_s \mathcal{L}'$.*

Minimal subsumption has the following natural properties.

Lemma 62. *Let $\mathcal{C}, \mathcal{D}$ be clauses. Then the following statements hold:*

(i) If $\mathcal{C} \leq_s \bot$, then $\mathcal{C} = \bot$.
(ii) If $\mathcal{C} \leq_s \mathcal{D}$, then $\models C \sqsubseteq D$.

As the following example shows, the subsumption lemma does not hold for $\mathfrak{T}$ and $\mathfrak{T}^u$. (Also see (Auffray, Enjalbert, and Hébrard 1990), page 16.)

Example 63. *One can derive the clause $\exists r.(A, B)$ from $\exists r.(A), \forall r.(\neg A \sqcup B)$, but from $\exists r.(A)$ and $\forall r.(B)$ (for which $\forall r.(B) \leq_s \forall r.(\neg B \sqcup C)$ holds), one cannot derive any clause neither in $\mathfrak{T}$ nor in $\mathfrak{T}^u$. One can derive $\exists r.(A, B)$ from $\exists r.(A)$ and $\forall r.(B)$, however, in $\mathfrak{T}^f$.*

In the following we write $\mathcal{C} = \mathcal{D}_1 \dot\sqcup \mathcal{D}_2$ if $\mathcal{C} = \mathcal{D}_1 \cup \mathcal{D}_2$ and $\mathcal{D}_1 \cap \mathcal{D}_2 = \emptyset$.

Lemma 64 (Subsumption Lemma)**.** *Let $\mathcal{C}, \mathcal{D}, \mathcal{E}$ be clauses such that $\mathcal{C}(,\mathcal{D}) \Rightarrow^f_\alpha \mathcal{E}$. Additionally, let $\mathcal{C}', \mathcal{D}'$ be clauses such that $\mathcal{C}' \leq_s \mathcal{C}$ and $\mathcal{D}' \leq_s \mathcal{D}$.*

Then one of the following propositions hold

- *$\mathcal{C}' \leq_s \mathcal{E}$, or*
- *$\mathcal{D}' \leq_s \mathcal{E}$, or*
- *there exists a $\mathfrak{T}^f$-derivation Δ' of a clause $\mathcal{E}'$ from $\{\mathcal{C}', \mathcal{D}'\}$ such that*
  - *$\mathcal{E}' \leq_s \mathcal{E}$,*
  - Inferences$(\Delta') \subseteq \{\alpha, \exists_f\}$,
  - *$\mathcal{C}' \in$* Leaves(Δ')*, and*
  - *$\mathcal{D}' \in$* Leaves(Δ') *if $\mathcal{C}, \mathcal{D} \Rightarrow_\alpha \mathcal{E}$.*

Proof. By induction on the depth of $\mathcal{C}(,\mathcal{D}) \Rightarrow_\alpha \mathcal{E}$.

We now distinguish between the different rules that were used to obtain the last derivation step of $\mathcal{C}_1(,\mathcal{C}_2) \Rightarrow_\alpha \mathcal{E}$.

- rule $\bot$: we have $\mathcal{C} = \mathcal{C}_1 \dot\sqcup \forall r.\bot$, $\mathcal{D} = \mathcal{D}_1 \dot\sqcup \exists r.(\mathcal{F})$, and $\mathcal{E} = \mathcal{C}_1 \sqcup \mathcal{D}_1$.
  If $\mathcal{C}' \leq_s \mathcal{C}_1$ or $\mathcal{D}' \leq_s \mathcal{D}_1$, we immediately obtain that $\mathcal{C}' \leq_s \mathcal{E}$ or $\mathcal{D}' \leq_s \mathcal{E}$ holds, respectively.
  We can now assume that $\mathcal{C}' = \tilde{\mathcal{C}}_1 \dot\sqcup \forall r.\bot$, $\mathcal{D}' = \tilde{\mathcal{D}}_1 \dot\sqcup \exists r.(\tilde{\mathcal{F}}^{(1)}) \dot\sqcup \ldots \dot\sqcup \exists r.(\tilde{\mathcal{F}}^{(m)})$ such that $m \geq 1$, $\tilde{\mathcal{C}}_1 \leq_s \mathcal{C}_1$, $\tilde{\mathcal{D}}_1 \leq_s \mathcal{D}_1$ and $\tilde{\mathcal{F}}^{(1)} \leq_s \mathcal{F}, \ldots, \tilde{\mathcal{F}}^{(m)} \leq_s \mathcal{F}$. It is then easy to see that there exists a derivation Δ' of the clause $\tilde{\mathcal{C}}_1 \sqcup \tilde{\mathcal{D}}_1$ such that Inferences$(\Delta') = \{\bot\}$, $\{\mathcal{C}', \mathcal{D}'\} \subseteq$ Leaves(Δ') and $\tilde{\mathcal{C}}_1 \sqcup \tilde{\mathcal{D}}_1 \leq_s \mathcal{C}_1 \sqcup \mathcal{D}_1 = \mathcal{E}$.
- rule A: can be proved similarly to $\bot$
- rule $\forall\exists$:
  we have $\mathcal{C} = \mathcal{C}_1 \dot\sqcup \forall r.\mathcal{C}_2$, $\mathcal{D} = \mathcal{D}_1 \dot\sqcup \exists r.(\mathcal{D}_2, \mathcal{F}_2)$, $\mathcal{E} = \mathcal{C}_1 \sqcup \mathcal{D}_1 \sqcup \exists r.(\mathcal{D}_2, \mathcal{F}_2, \mathcal{E}_2)$, and $\mathcal{C}_2, \mathcal{D}_2 \Rightarrow_\alpha \mathcal{E}_2$.
  If $\mathcal{C}' \leq_s \mathcal{C}_1$ or $\mathcal{D}' \leq_s \mathcal{D}_1$, we immediately obtain that $\mathcal{C}' \leq_s \mathcal{E}$ or $\mathcal{D}' \leq_s \mathcal{E}$ holds, respectively. Otherwise, let
  $$\mathcal{C}' = \tilde{\mathcal{C}}_1 \dot\sqcup \forall r.\tilde{\mathcal{C}}_2^{(1)} \dot\sqcup \ldots \dot\sqcup \forall r.\tilde{\mathcal{C}}_2^{(m)}$$
  and
  $$\mathcal{D}' = \tilde{\mathcal{D}}_1 \dot\sqcup \exists r.(\tilde{\mathcal{D}}_2^{(1)}, \tilde{\mathcal{F}}^{(1)}) \dot\sqcup \ldots \dot\sqcup \exists r.(\tilde{\mathcal{D}}_2^{(n)}, \tilde{\mathcal{F}}^{(n)})$$
  such that $\tilde{\mathcal{C}}_1 \leq_s \mathcal{C}_1$, $\tilde{\mathcal{D}}_1 \leq_s \mathcal{D}_1$, $m \geq 1$, $n \geq 1$, $\tilde{\mathcal{C}}_2^{(i)} \leq_s \mathcal{C}_2$ for every $1 \leq i \leq m$, $\tilde{\mathcal{D}}_2^{(j)} \leq_s \mathcal{D}_2$ for every $1 \leq j \leq n$, and $\{\tilde{\mathcal{D}}_2^{(k)}, \tilde{\mathcal{F}}^{(k)}\} \leq_s \{\mathcal{D}_2, \mathcal{F}\}$ for every $1 \leq k \leq n$.
  It then follows from the induction hypothesis for every $1 \leq i \leq m$ and for every $1 \leq j \leq n$ that either
  - $\tilde{\mathcal{C}}_2^{(i)} \leq_s \mathcal{E}_2$, or

– $\tilde{\mathcal{D}}_2^{(j)} \leq_s \mathcal{E}_2$, or
– there exists a $\mathfrak{T}^f$-derivation $\Gamma^{(i,j)}$ of a clause $\tilde{\mathcal{E}}^{(i,j)}$ from $\{\tilde{\mathcal{C}}_2^{(i)}, \tilde{\mathcal{D}}_2^{(j)}\}$ such that
  * $\tilde{\mathcal{E}}^{(i,j)} \leq_s \mathcal{E}_2$,
  * $\text{Inferences}(\Gamma^{(i,j)}) \subseteq \{\alpha, \exists_f\}$, and
  * $\tilde{\mathcal{C}}_2^{(i)}, \tilde{\mathcal{D}}_2^{(j)} \in \text{Leaves}(\Gamma^{(i,j)})$.

Thus, if $\tilde{\mathcal{D}}_2^{(j)} \leq_s \mathcal{E}_2$ holds for every $1 \leq j \leq m$, we can immediately conclude that $\mathcal{D}' \leq_s \mathcal{E}$ holds. Now, let $\emptyset \neq \{j_1, \ldots, j_l\} = \{\, j \mid 1 \leq j \leq n \text{ and } \tilde{\mathcal{D}}_2^{(j)} \not\leq_s \mathcal{E}_2 \,\}$ and $\tilde{\mathcal{D}}_1^{\exists} = \tilde{\mathcal{D}}_1 \sqcup \bigsqcup_{j \notin \{j_1,\ldots,j_l\}} \exists r.(\tilde{\mathcal{D}}_2^{(j)}, \tilde{\mathcal{F}}^{(j)})$. Then, if $\tilde{\mathcal{C}}_2^{(1)} \leq_s \mathcal{E}_2$, we can derive the clause

$$\begin{aligned} X_{(j_1,1)} = {}& \tilde{\mathcal{C}}_1 \sqcup \tilde{\mathcal{D}}_1^{\exists} \\ & \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)}, \tilde{\mathcal{C}}_2^{(1)}) \qquad \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_2)}, \tilde{\mathcal{F}}^{(j_2)}) \sqcup \ldots \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_l)}, \tilde{\mathcal{F}}^{(j_l)}) \\ & \sqcup \forall r.\tilde{\mathcal{C}}_2^{(2)} \sqcup \ldots \sqcup \forall r.\tilde{\mathcal{C}}_2^{(m)} \end{aligned}$$

from $\{\mathcal{C}', \mathcal{D}'\}$ by using the $(\exists_f)$ inference rule and such that $\{\mathcal{C}', \mathcal{D}'\} \subseteq \text{Leaves}(\Delta^{(1,1)})$.
Otherwise, by Lemmata 44 and 46 there exists a $\mathfrak{T}$-derivation $\delta^{(j_1,1)}$ of the clause

$$\exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)}, \tilde{\mathcal{E}}_2^{(j_1,1)}, \tilde{\mathcal{G}}^{(j_1,1)})$$

from $\{\forall r.\tilde{\mathcal{C}}_2^{(1)}, \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)})\}$ such that $\text{Inferences}(\delta^{(j_1,1)}) \subseteq \{\alpha, \exists_f\}$ and $\{\forall r.\tilde{\mathcal{C}}_2^{(1)}, \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)})\} \subseteq \text{Leaves}(\delta^{(j_1,1)})$. It it thus easy to see that there exists a derivation $\Delta^{(j_1,1)}$ of the clause

$$\begin{aligned} X_{(j_1,1)} = {}& \tilde{\mathcal{C}}_1 \sqcup \tilde{\mathcal{D}}_1^{\exists} \\ & \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)}, \tilde{\mathcal{E}}_2^{(j_1,1)}, \tilde{\mathcal{G}}^{(j_1,1)}) \qquad \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_2)}, \tilde{\mathcal{F}}^{(j_2)}) \sqcup \ldots \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_l)}, \tilde{\mathcal{F}}^{(j_l)}) \\ & \sqcup \forall r.\tilde{\mathcal{C}}_2^{(2)} \sqcup \ldots \sqcup \forall r.\tilde{\mathcal{C}}_2^{(m)} \end{aligned}$$

from $\{\mathcal{C}', \mathcal{D}'\}$ such that $\text{Inferences}(\Delta^{(j_1,1)}) \subseteq \{\alpha, \exists_f\}$ and $\{\mathcal{C}', \mathcal{D}'\} \subseteq \text{Leaves}(\Delta^{(j_1,1)})$.
By using similar arguments iteratively on the pairs of literals

$$(\forall r.\tilde{\mathcal{C}}_2^{(1)}, \exists r.(\tilde{\mathcal{D}}_2^{(j_2)}, \tilde{\mathcal{F}}^{(2)}), \ldots, (\forall r.\tilde{\mathcal{C}}_2^{(1)}, \exists r.(\tilde{\mathcal{D}}_2^{(n)}, \tilde{\mathcal{F}}^{(n)}),$$

one can show that there exists a $\mathfrak{T}$-derivation $\Delta^{(j_l,1)}$ of the clause

$$\begin{aligned} X_{(j_l,1)} = {}& \tilde{\mathcal{C}}_1 \sqcup \tilde{\mathcal{D}}_1^{\exists} \\ & \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)}, \tilde{\mathcal{E}}_2^{(j_1,1)}, \tilde{\mathcal{G}}^{(j_1,1)}) \sqcup \ldots \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_l)}, \tilde{\mathcal{F}}^{(j_l)}, \tilde{\mathcal{E}}_2^{(j_l,1)}, \tilde{\mathcal{G}}^{(j_l,1)}) \\ & \sqcup \forall r.\tilde{\mathcal{C}}_2^{(2)} \sqcup \ldots \sqcup \forall r.\tilde{\mathcal{C}}_2^{(m)} \end{aligned}$$

from $\{\mathcal{C}', X_{(j_1,1)}\}$ such that $\text{Inferences}(\Delta^{(j_l,1)}) \subseteq \{\alpha, \exists_f\}$. Note that for some $j \in \{j_1, \ldots, j_l\}$ it can hold that $\tilde{\mathcal{E}}_2^{(j,1)} = \tilde{\mathcal{C}}_2^{(1)}$ and $\tilde{\mathcal{G}}^{(j,1)} = \emptyset$. We can continue with the pair of literals $(\forall r.\tilde{\mathcal{C}}_2^{(2)}, \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)})$ and we obtain a $\mathfrak{T}$-derivation $\Delta^{(j_1,2)}$ of the clause

$$\begin{aligned} X_{(j_1,2)} = {}& \tilde{\mathcal{C}}_1 \sqcup \tilde{\mathcal{D}}_1^{\exists} \\ & \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)}, \tilde{\mathcal{E}}_2^{(j_1,1)}, \tilde{\mathcal{G}}^{(j_1,1)}) \sqcup \ldots \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_l)}, \tilde{\mathcal{F}}^{(j_l)}, \tilde{\mathcal{E}}_2^{(j_l,1)}, \tilde{\mathcal{G}}^{(j_l,1)}) \\ & \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)}, \tilde{\mathcal{E}}_2^{(j_1,2)}, \tilde{\mathcal{G}}^{(j_1,2)}) \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_2)}, \tilde{\mathcal{F}}^{(j_2)}) \sqcup \ldots \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_l)}, \tilde{\mathcal{F}}^{(j_l)}) \\ & \sqcup \forall r.\tilde{\mathcal{C}}_2^{(3)} \sqcup \ldots \sqcup \forall r.\tilde{\mathcal{C}}_2^{(m)} \end{aligned}$$

from $\{\mathcal{D}', X_{(n,1)}\}$ such that $\text{Inferences}(\Delta^{(j_1,2)}) \subseteq \{\alpha, \exists_f\}$. Applying similar arguments on the pairs of literals

$$(\forall r.\tilde{\mathcal{C}}_2^{(2)}, \exists r.(\tilde{\mathcal{D}}_2^{(j_2)}, \tilde{\mathcal{F}}^{(j_2)}), \ldots, (\forall r.\tilde{\mathcal{C}}_2^{(2)}, \exists r.(\tilde{\mathcal{D}}_2^{(j_l)}, \tilde{\mathcal{F}}^{(j_l)}),$$

allows us to obtain a $\mathfrak{T}$-derivation $\Delta^{(j_l,2)}$ of the clause

$$\begin{aligned} X_{(j_l,2)} = {}& \tilde{\mathcal{C}}_1 \sqcup \tilde{\mathcal{D}}_1^{\exists} \\ & \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)}, \tilde{\mathcal{E}}_2^{(j_1,1)}, \tilde{\mathcal{G}}^{(j_1,1)}) \sqcup \ldots \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_l)}, \tilde{\mathcal{F}}^{(j_l)}, \tilde{\mathcal{E}}_2^{(j_l,1)}, \tilde{\mathcal{G}}^{(j_l,1)}) \\ & \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)}, \tilde{\mathcal{E}}_2^{(j_1,2)}, \tilde{\mathcal{G}}^{(j_1,2)}) \sqcup \ldots \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_l)}, \tilde{\mathcal{F}}^{(j_l)}, \tilde{\mathcal{E}}_2^{(j_l,2)}, \tilde{\mathcal{G}}^{(j_l,2)}) \\ & \sqcup \forall r.\tilde{\mathcal{C}}_2^{(3)} \sqcup \ldots \sqcup \forall r.\tilde{\mathcal{C}}_2^{(m)} \end{aligned}$$

from $\{X_{j_k,1}, X_{j_1,2}\}$ such that $\mathrm{Inferences}(\Delta^{(j_l,2)}) \subseteq \{\alpha, \exists_f\}$. Finally, one can show analogously that there exists a $\mathfrak{T}$-derivation $\Delta^{(j_l,m)}$ of the clause

$$\begin{aligned}X_{(j_l,m)} = & \tilde{\mathcal{C}}_1 \sqcup \tilde{\mathcal{D}}_1^{\exists} \\ & \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)}, \tilde{\mathcal{E}}_2^{(j_1,1)}, \tilde{\mathcal{G}}^{(j_1,1)}) \sqcup \ldots \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_l)}, \tilde{\mathcal{F}}^{(j_l)}, \tilde{\mathcal{E}}_2^{(j_l,1)}, \tilde{\mathcal{G}}^{(j_l,1)}) \\ & \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)}, \tilde{\mathcal{E}}_2^{(j_1,2)}, \tilde{\mathcal{G}}^{(j_1,2)}) \sqcup \ldots \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_l)}, \tilde{\mathcal{F}}^{(j_l)}, \tilde{\mathcal{E}}_2^{(j_l,2)}, \tilde{\mathcal{G}}^{(j_l,2)}) \\ & \sqcup \ldots \\ & \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_1)}, \tilde{\mathcal{F}}^{(j_1)}, \tilde{\mathcal{E}}_2^{(j_1,m)}, \tilde{\mathcal{G}}^{(j_1,m)}) \sqcup \ldots \sqcup \exists r.(\tilde{\mathcal{D}}_2^{(j_l)}, \tilde{\mathcal{F}}^{(j_l)}, \tilde{\mathcal{E}}_2^{(j_l,m)}, \tilde{\mathcal{G}}^{(j_l,m)})\end{aligned}$$

from $\{\mathcal{D}', X_{(j_l,2)}\}$ such that $\mathrm{Inferences}(\Delta^{(j_l,m)}) \subseteq \{\alpha, \exists_f\}$. For $\mathcal{E}' = X_{(j_l,m)}$ it is easy to see that $\mathcal{E}' \leq_s \mathcal{E}$ holds.

- rule $\exists_f$: can be proved similarly to $\forall\exists$
- rule $\forall\forall$: can be proved similarly to $\forall\exists$
- rule $\exists_2$: we have $\mathcal{C} = \mathcal{C}_1 \dot\sqcup \exists r.(\mathcal{C}_2, \mathcal{D}_2, \mathcal{F})$, $\mathcal{E} = \mathcal{C}_1 \sqcup \exists r.(\mathcal{C}_2, \mathcal{D}_2, \mathcal{F}, \mathcal{E}_2)$ and $\mathcal{C}_2, \mathcal{D}_2 \Rightarrow_\alpha \mathcal{E}_2$
  If $\mathcal{C}' \leq_s \mathcal{C}_1$, we immediately obtain that $\mathcal{C}' \leq_s \mathcal{E}$ holds. Otherwise, let

$$\mathcal{C}' = \tilde{\mathcal{C}}_1 \dot\sqcup \exists r.(\tilde{\mathcal{C}}_2^{(1)}, \tilde{\mathcal{D}}_2^{(1)}, \tilde{\mathcal{F}}^{(1)}) \dot\sqcup \ldots \dot\sqcup \exists r.(\tilde{\mathcal{C}}_2^{(m)}, \tilde{\mathcal{D}}_2^{(m)}, \tilde{\mathcal{F}}^{(m)})$$

  such that $\tilde{\mathcal{C}}_1 \leq_s \mathcal{C}_1$, $\tilde{\mathcal{D}}_1 \leq_s \mathcal{D}_1$, $m \geq 1$, $\tilde{\mathcal{C}}_2^{(i)} \leq_s \mathcal{C}_2$, $\tilde{\mathcal{D}}_2^{(i)} \leq_s \mathcal{D}_2$, and $\{\tilde{\mathcal{C}}_2^{(i)}, \tilde{\mathcal{D}}_2^{(i)}, \tilde{\mathcal{F}}^{(i)}\} \leq_s \{\mathcal{C}_2, \mathcal{D}_2, \mathcal{F}\}$ for every $1 \leq i \leq m$. It then follows from the induction hypothesis for every $1 \leq i \leq m$ that either
  - $\tilde{\mathcal{C}}_2^{(i)} \leq_s \mathcal{E}_2$, or
  - $\tilde{\mathcal{D}}_2^{(i)} \leq_s \mathcal{E}_2$, or
  - there exists a $\mathfrak{T}^f$-derivation $\Gamma^{(i)}$ of a clause $\tilde{\mathcal{E}}^{(i)}$ from $\{\tilde{\mathcal{C}}_2^{(i)}, \tilde{\mathcal{D}}_2^{(i)}\}$ such that
    - $\tilde{\mathcal{E}}^{(i)} \leq_s \mathcal{E}_2$,
    - $\mathrm{Inferences}(\Gamma^{(i)}) \subseteq \{\alpha, \exists_f\}$, and
    - $\tilde{\mathcal{C}}_2^{(i)}, \tilde{\mathcal{D}}_2^{(i)} \in \mathrm{Leaves}(\Gamma^{(i)})$.

  Thus, if $\tilde{\mathcal{C}}_2^{(i)} \leq_s \mathcal{E}_2$ or if $\tilde{\mathcal{D}}_2^{(i)} \leq_s \mathcal{E}_2$ holds for every $1 \leq i \leq m$, we can immediately conclude that $\mathcal{D}' \leq_s \mathcal{E}$ holds. Now, let $\emptyset \neq \{i_1, \ldots, i_l\} = \{\, i \mid 1 \leq i \leq m, \tilde{\mathcal{C}}_2^{(i)} \not\leq_s \mathcal{E}_2 \text{ and } \tilde{\mathcal{D}}_2^{(i)} \not\leq_s \mathcal{E}_2 \,\}$ and $\tilde{\mathcal{C}}_1^{\exists} = \tilde{\mathcal{C}}_1 \sqcup \bigsqcup_{i \notin \{i_1,\ldots,i_l\}} \exists r.(\tilde{\mathcal{D}}_2^{(i)}, \tilde{\mathcal{F}}^{(i)})$. By applying Lemma 46 iteratively, one can show that there exists a $\mathfrak{T}$-derivation $\Delta^{(i_l)}$ of the clause

$$\begin{aligned}X_{(i_l)} = & \tilde{\mathcal{C}}_1^{\exists} \\ & \sqcup \exists r.(\tilde{\mathcal{C}}_2^{(i_1)}, \tilde{\mathcal{D}}_2^{(i_1)}, \tilde{\mathcal{F}}^{(i_1)}, \tilde{\mathcal{E}}_2^{(i_1)}, \tilde{\mathcal{G}}^{(i_1)}) \\ & \sqcup \ldots \\ & \sqcup \exists r.(\tilde{\mathcal{C}}_2^{(i_l)}, \tilde{\mathcal{D}}_2^{(i_l)}, \tilde{\mathcal{F}}^{(i_l)}, \tilde{\mathcal{E}}_2^{(i_l)}, \tilde{\mathcal{G}}^{(i_l)})\end{aligned}$$

  from $\{\mathcal{C}'\}$ such that $\tilde{\mathcal{E}}_2^{(i)} \leq_s$ and $\mathrm{Inferences}(\Delta^{(i_l)}) \subseteq \{\alpha, \exists_f\}$. For $\mathcal{E}' = X_{(i_l)}$ it is easy to see that $\mathcal{E}' \leq_s \mathcal{E}$ holds.
- rule $\exists_1$: can be proved similarly to $\exists_2$
- rule $\forall$: can be proved similarly to $\exists_1$

□

The operation $\mathrm{Supp}(\Upsilon, \mathcal{C})$, which is used to remove/simplify clauses $\mathcal{C}$ that contain unwanted concept names from a signature Υ, is defined as follows. (The following definition is equivalent to the definition given in (Herzig and Mengin 2008).)

Definition 65. *For a clause $\mathcal{C}$ (a set of clauses $\mathcal{N}$) and signature $\Upsilon \subseteq \mathsf{N_C}$ let $\mathrm{Supp}(\Upsilon, \mathcal{C})$ ($\mathrm{Supp}(\Upsilon, \mathcal{N})$) denote the (set of) clause(s) or $\top$ that results from $\mathcal{C}$ (from $\mathcal{N}$) by exhaustively applying the following rewrite rules:*

- $\mathcal{D} \sqcup A \to \top$ *and* $\mathcal{D} \sqcup \neg A \to \top$ *for* $A \in \Upsilon$
- $\forall r.\top \to \top$
- $\mathcal{D} \sqcup \top \to \top$ *for a clause* $\mathcal{D}$
- $\mathcal{F} \cup \{\top\} \to \mathcal{F}$ *for a set of clauses* $\mathcal{F}$

Lemma 66. *Let $\Upsilon \subseteq \mathsf{N_C}$ be a signature and let $\mathcal{C}, \mathcal{D}, \mathcal{E}$ be clauses such that $\mathcal{C}(,\mathcal{D}) \Rightarrow_\alpha^f \mathcal{E}$ and $\alpha = \bot$ or $\alpha \notin \Upsilon$.*
Then one of the following holds:

(a) $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$ *and* $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$*, or*

(b) $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$ *and* $\mathrm{Supp}(\Upsilon, \mathcal{D}) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E})$*, or*

(c) $\mathrm{Supp}(\Upsilon, \mathcal{D}) = \top$ *and* $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$*, or*

(d) $\mathrm{Supp}(\Upsilon, \mathcal{D}) = \top$ *and* $\mathrm{Supp}(\Upsilon, \mathcal{C}) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E})$*, or*

(e) $\mathrm{Supp}(\Upsilon, \mathcal{C}) \neq \top$*,* $\mathrm{Supp}(\Upsilon, \mathcal{D}) \neq \top$ *and* $\mathrm{Supp}(\Upsilon, \mathcal{C}), \mathrm{Supp}(\Upsilon, \mathcal{D}) \Rightarrow^f_\alpha \mathcal{E}'$ *with* $\mathcal{E}' \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E})$ *and* $\mathrm{Depth}(\mathcal{E}') \leq \mathrm{Depth}(\mathcal{E})$.

Proof. By induction on the structure of $\mathcal{C}, \mathcal{D} \Rightarrow_\alpha \mathcal{E}$.

We now distinguish between the different rules that were used to obtain the last derivation step of $\mathcal{C}_1(,\mathcal{C}_2) \Rightarrow_\alpha \mathcal{E}$.

- rule A: we have $\mathcal{C} = \mathcal{C} \setminus \{A\} \sqcup A$, $\mathcal{D} = \mathcal{D} \setminus \{\neg A\} \sqcup \neg A$, and $\mathcal{E} = \mathcal{C} \setminus \{A\} \sqcup \mathcal{D} \setminus \{\neg A\}$.
  If $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{A\}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\neg A\}) = \top$, we immediately obtain that $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$, and $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D}) = \top$.
  We can now assume that $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{A\}) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\neg A\}) \neq \top$. As $A \notin \Upsilon$, we have $\mathrm{Supp}(\Upsilon, \mathcal{C}) \neq \top$, $\mathrm{Supp}(\Upsilon, \mathcal{D}) \neq \top$, and
$$\begin{aligned}\mathrm{Supp}(\Upsilon, \mathcal{C}), \mathrm{Supp}(\Upsilon, \mathcal{D}) \Rightarrow_A &\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{A\}) \sqcup \mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\neg A\}) \\ &= \mathrm{Supp}(\Upsilon, \mathcal{E}).\end{aligned}$$

- rule $\bot$: we have $\mathcal{C} = \mathcal{C} \setminus \{\exists r.(\mathcal{F})\} \sqcup \exists r.(\mathcal{F})$, $\mathcal{D} = \mathcal{D} \setminus \{\forall r.\bot\} \sqcup \forall r.\bot$, and $\mathcal{E} = \mathcal{C} \setminus \{\exists r.(\mathcal{F})\} \sqcup \mathcal{D} \setminus \{\forall r.\bot\}$.
  If $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{F})\}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\forall r.\bot\}) = \top$, we immediately obtain that $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$, and $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D}) = \top$.
  We can now assume that $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{F})\}) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\forall r.\bot\}) \neq \top$. As $\mathrm{Supp}(\Upsilon, \exists r.(\mathcal{F})) \neq \top$ and $\mathrm{Supp}(\Upsilon, \forall r.\bot) \neq \top$, we have $\mathrm{Supp}(\Upsilon, \mathcal{C}) \neq \top$, $\mathrm{Supp}(\Upsilon, \mathcal{D}) \neq \top$, and
$$\begin{aligned}\mathrm{Supp}(\Upsilon, \mathcal{C}), \mathrm{Supp}(\Upsilon, \mathcal{D}) \Rightarrow_\bot &\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{F})\}) \sqcup \mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\forall r.\bot\}) \\ &= \mathrm{Supp}(\Upsilon, \mathcal{E}).\end{aligned}$$

- rule $\exists_f$: we have $\mathcal{C} = \mathcal{C} \setminus \{\exists r.(\mathcal{F}_2)\} \sqcup \exists r.(\mathcal{F}_2)$, $\mathcal{D} = \mathcal{D} \setminus \{\forall r.\mathcal{D}_2\} \sqcup \forall r.\mathcal{D}_2$, and $\mathcal{E} = \mathcal{C} \setminus \{\exists r.(\mathcal{F}_2)\} \sqcup \mathcal{D} \setminus \{\forall r.\mathcal{D}_2\} \sqcup \exists r.(\mathcal{F}, \mathcal{D}_2)$.
  If $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{F}_2)\}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\forall r.\mathcal{D}_2\}) = \top$, we immediately obtain that $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$, and $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D}) = \top$.
  We can now assume that $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{F}_2)\}) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\forall r.\mathcal{D}_2\}) \neq \top$.
  If $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) = \top$, we can infer that $\mathrm{Supp}(\Upsilon, \mathcal{D}) = \top$ and
$$\begin{aligned}\mathrm{Supp}(\Upsilon, \mathcal{C}) &= \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{F}_2)\}) \sqcup \exists r.(\mathrm{Supp}(\mathcal{F}_2)) \\ &\leq_s \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{F}_2)\} \sqcup \mathcal{D} \setminus \{\forall r.\mathcal{D}_2\}) \sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{D}_2), \mathrm{Supp}(\mathcal{F}_2)) \\ &= \mathrm{Supp}(\Upsilon, \mathcal{E})\end{aligned}$$
  We can now assume that $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \neq \top$. Thus, as $\mathrm{Supp}(\Upsilon, \exists r.(\mathcal{F}_2)) \neq \top$ and $\mathrm{Supp}(\Upsilon, \forall r.\mathcal{D}_2) \neq \top$, we have $\mathrm{Supp}(\Upsilon, \mathcal{C}) \neq \top$, $\mathrm{Supp}(\Upsilon, \mathcal{D}) \neq \top$, and
$$\begin{aligned}\mathrm{Supp}(\Upsilon, \mathcal{C}), \mathrm{Supp}(\Upsilon, \mathcal{D}) \Rightarrow_{\exists_f} &\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{F}_2)\}) \sqcup \mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\forall r.\mathcal{D}_2\}) \\ &\sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{F}_2), \mathrm{Supp}(\Upsilon, \mathcal{D}_2)) \\ &= \mathrm{Supp}(\Upsilon, \mathcal{E}).\end{aligned}$$

- rule $\forall\exists$:
  we have $\mathcal{C} = \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \forall r.\mathcal{C}_2$, $\mathcal{D} = \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\} \sqcup \exists r.(\mathcal{D}_2, \mathcal{F}_2)$, $\mathcal{E} = \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\} \sqcup \exists r.(\mathcal{D}_2, \mathcal{F}_2, \mathcal{E}_2)$, and $\mathcal{C}_2, \mathcal{D}_2 \Rightarrow_\alpha \mathcal{E}_2$.
  If $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\}) = \top$, we immediately obtain that $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$, and $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D}) = \top$.
  We can now assume that $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\}) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\}) \neq \top$. It follows from the induction hypothesis that either
  (a) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$, or
  (b) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, or
  (c) $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$, or
  (d) $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, or
  (e) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$, $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}_2), \mathrm{Supp}(\Upsilon, \mathcal{D}_2) \Rightarrow^f_\alpha \mathcal{E}''$ with $\mathcal{E}'' \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$.

If $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$, we obtain $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, i.e.

$$\begin{aligned}
\mathrm{Supp}(\Upsilon, \mathcal{D}) &= \mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\}) \sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{D}_2), \mathrm{Supp}(\mathcal{F}_2)) \\
&\leq_s \mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\}) \sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{D}_2), \mathrm{Supp}(\Upsilon, \mathcal{E}_2), \mathrm{Supp}(\mathcal{F}_2)) \\
&\leq_s \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\}) \sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{D}_2), \mathrm{Supp}(\Upsilon, \mathcal{E}_2), \mathrm{Supp}(\mathcal{F}_2)) \\
&= \mathrm{Supp}(\Upsilon, \mathcal{E})
\end{aligned}$$

Now, we assume that $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$.
If $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) = \top$, it holds that $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$. Hence, we can derive

$$\begin{aligned}
\mathrm{Supp}(\Upsilon, \mathcal{C}), \mathrm{Supp}(\Upsilon, \mathcal{D}) \Rightarrow^s_\alpha &\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\}) \\
&\sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{C}_2), \mathrm{Supp}(\Upsilon, \mathcal{E}_2), \mathrm{Supp}(\mathcal{F}_2)) \\
&\leq_s \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\}) \\
&\sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{D}_2), \mathrm{Supp}(\Upsilon, \mathcal{E}_2), \mathrm{Supp}(\mathcal{F}_2)) \\
&= \mathrm{Supp}(\Upsilon, \mathcal{E})
\end{aligned}$$

Now, we assume that $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \neq \top$. Thus, we can derive

$$\begin{aligned}
\mathrm{Supp}(\Upsilon, \mathcal{C}), \mathrm{Supp}(\Upsilon, \mathcal{D}) \Rightarrow^f_\alpha &\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\}) \\
&\sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{D}_2), \mathcal{E}', \mathrm{Supp}(\mathcal{F}_2)) \\
&\leq_s \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\}) \\
&\sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{D}_2), \mathrm{Supp}(\Upsilon, \mathcal{E}_2), \mathrm{Supp}(\mathcal{F}_2)) \\
&= \mathrm{Supp}(\Upsilon, \mathcal{E})
\end{aligned}$$

- rule $\forall\forall$ we have $\mathcal{C} = \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \forall r.\mathcal{C}_2$, $\mathcal{D} = \mathcal{D} \setminus \{\forall r.\mathcal{D}_2\} \sqcup \forall r.\mathcal{D}_2$, $\mathcal{E} = \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \mathcal{D} \setminus \{\forall r.\mathcal{D}_2\} \sqcup \forall r.\mathcal{E}_2$, and $\mathcal{C}_2, \mathcal{D}_2 \Rightarrow_\alpha \mathcal{E}_2$.
If $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\forall r.\mathcal{D}_2\}) = \top$, we immediately obtain that $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$, and $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D}) = \top$.
We can now assume that $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\}) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{D} \setminus \{\forall r.\mathcal{D}_2\}) \neq \top$. It follows from the induction hypothesis that either

(a) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$, or
(b) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, or
(c) $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$, or
(d) $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, or
(e) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$, $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}_2), \mathrm{Supp}(\Upsilon, \mathcal{D}_2) \Rightarrow^f_\alpha \mathcal{E}''$ with $\mathcal{E}'' \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$.

If $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$, we obtain $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, i.e. either $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D}) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E})$ holds.
Similarly, if $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) = \top$, we obtain $\mathrm{Supp}(\Upsilon, \mathcal{D}) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, i.e. either $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{C}) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E})$ holds.
We can now assume that $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \neq \top$, i.e. we can derive

$$\begin{aligned}
\mathrm{Supp}(\Upsilon, \mathcal{C}), \mathrm{Supp}(\Upsilon, \mathcal{D}) \Rightarrow^f_\alpha &\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \mathcal{D} \setminus \{\forall r.\mathcal{D}_2\}) \\
&\sqcup \forall r.\mathcal{E}'' \\
&\leq_s \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \mathcal{D} \setminus \{\exists r.(\mathcal{D}_2, \mathcal{F}_2)\}) \\
&\sqcup \forall r.\mathrm{Supp}(\Upsilon, \mathcal{E}_2) \\
&= \mathrm{Supp}(\Upsilon, \mathcal{E})
\end{aligned}$$

□

Lemma 67. *Let $\Upsilon \subseteq \mathsf{N_C}$ be a signature and let $\mathcal{C}, \mathcal{E}$ be clauses such that $\mathcal{C} \Rightarrow^f_\alpha \mathcal{E}$ and $\alpha = \bot$ or $\alpha \notin \Upsilon$. Then one of the following holds:*

(a) $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$ *and* $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$*, or*
(b) $\mathrm{Supp}(\Upsilon, \mathcal{C}) \neq \top$ *and* $\mathrm{Supp}(\Upsilon, \mathcal{C}) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E})$*, or*
(c) $\mathrm{Supp}(\Upsilon, \mathcal{C}) \neq \top$ *and* $\mathrm{Supp}(\Upsilon, \mathcal{C}) \Rightarrow^f_\alpha \mathcal{E}'$ *with* $\mathcal{E}' \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E})$ *and* $\mathrm{Depth}(\mathcal{E}') \leq \mathrm{Depth}(\mathcal{E})$.

Proof. By induction on the structure of $\mathcal{C} \Rightarrow_\alpha \mathcal{E}$.

- rule $\exists_2$: we have $\mathcal{C} = \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{D}_2, \mathcal{F})\} \sqcup \exists r.(\mathcal{C}_2, \mathcal{D}_2, \mathcal{F})$, $\mathcal{E} = \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{D}_2 \mathcal{F})\} \sqcup \exists r.(\mathcal{C}_2, \mathcal{D}_2, \mathcal{F}, \mathcal{E}_2)$ and $\mathcal{C}_2, \mathcal{D}_2 \Rightarrow_\alpha \mathcal{E}_2$.
  If $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{D}_2, \mathcal{F})\}) = \top$, we immediately obtain that $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$.
  We can now assume that $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{D}_2, \mathcal{F})\}) \neq \top$. It follows from Lemma 66 that either
  (a) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$, or
  (b) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, or
  (c) $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$, or
  (d) $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, or
  (e) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$, $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}_2), \mathrm{Supp}(\Upsilon, \mathcal{D}_2) \Rightarrow^f_\alpha \mathcal{E}''$ with $\mathcal{E}'' \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$.
  If $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$, we obtain $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$ or $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, i.e.
$$\begin{aligned}\mathrm{Supp}(\Upsilon, \mathcal{C}) &= \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{D}_2, \mathcal{F}_2)\}) \sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{D}_2), \mathrm{Supp}(\mathcal{F}_2))\\ &\leq_s \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{D}_2, \mathcal{F}_2)\}) \sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{D}_2), \mathrm{Supp}(\Upsilon, \mathcal{E}_2), \mathrm{Supp}(\mathcal{F}_2))\\ &= \mathrm{Supp}(\Upsilon, \mathcal{E})\end{aligned}$$
  The case for $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) = \top$ can be proved analogously. In the following we therefore assume that $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{D}_2) \neq \top$. Thus, we can derive
$$\begin{aligned}\mathrm{Supp}(\Upsilon, \mathcal{C}) \Rightarrow^f_\alpha &\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{D}_2, \mathcal{F}_2)\})\\ &\sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{C}_2), \mathrm{Supp}(\Upsilon, \mathcal{D}_2), \mathcal{E}'', \mathrm{Supp}(\mathcal{F}_2))\\ &\leq_s \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{D}_2, \mathcal{F}_2)\})\\ &\sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{C}_2), \mathrm{Supp}(\Upsilon, \mathcal{D}_2), \mathrm{Supp}(\Upsilon, \mathcal{E}_2), \mathrm{Supp}(\mathcal{F}_2))\\ &= \mathrm{Supp}(\Upsilon, \mathcal{E})\end{aligned}$$
- rule $\exists_1$: we have $\mathcal{C} = \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{F})\} \sqcup \exists r.(\mathcal{C}_2, \mathcal{F})$, $\mathcal{E} = \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{F})\} \sqcup \exists r.(\mathcal{C}_2, \mathcal{F}, \mathcal{E}_2)$ and $\mathcal{C}_2 \Rightarrow_\alpha \mathcal{E}_2$.
  If $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{F})\}) = \top$, we immediately obtain that $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$.
  We can now assume that $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{F})\}) \neq \top$. It follows from the induction hypothesis that either
  (a) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$, or
  (b) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, or
  (c) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \Rightarrow^f_\alpha \mathcal{E}''$ with $\mathcal{E}'' \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$ and $\mathrm{Depth}(\mathcal{E}_2') \leq \mathrm{Depth}(\mathcal{E}_2)$.
  If $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$, we obtain $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$, i.e.
$$\begin{aligned}\mathrm{Supp}(\Upsilon, \mathcal{C}) &= \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{F}_2)\}) \sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{F}_2))\\ &= \mathrm{Supp}(\Upsilon, \mathcal{E})\end{aligned}$$
  We now assume that $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$, i.e. either $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$ or $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \Rightarrow^f_\alpha \mathcal{E}''$ with $\mathcal{E}'' \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$ and $\mathrm{Depth}(\mathcal{E}_2') \leq \mathrm{Depth}(\mathcal{E}_2)$ holds.
  If $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, we can infer that
$$\begin{aligned}\mathrm{Supp}(\Upsilon, \mathcal{C}) &= \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{F}_2)\}) \sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{C}_2), \mathrm{Supp}(\Upsilon, \mathcal{F}_2))\\ &\leq_s \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{F}_2)\}) \sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{C}_2), \mathrm{Supp}(\Upsilon, \mathcal{F}_2), \mathrm{Supp}(\Upsilon, \mathcal{E}_2))\\ &= \mathrm{Supp}(\Upsilon, \mathcal{E})\end{aligned}$$
  If $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \Rightarrow^f_\alpha \mathcal{E}''$ with $\mathcal{E}'' \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$ and $\mathrm{Depth}(\mathcal{E}_2') \leq \mathrm{Depth}(\mathcal{E}_2)$, we can derive
$$\begin{aligned}\mathrm{Supp}(\Upsilon, \mathcal{C}) \Rightarrow^f_\alpha &\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{F}_2)\})\\ &\sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{C}_2), \mathcal{E}'', \mathrm{Supp}(\mathcal{F}_2))\\ &\leq_s \mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\exists r.(\mathcal{C}_2, \mathcal{F}_2)\})\\ &\sqcup \exists r.(\mathrm{Supp}(\Upsilon, \mathcal{C}_2), \mathrm{Supp}(\Upsilon, \mathcal{E}_2), \mathrm{Supp}(\mathcal{F}_2))\\ &= \mathrm{Supp}(\Upsilon, \mathcal{E})\end{aligned}$$
- rule $\forall_1$: we have $\mathcal{C} = \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \forall r.\mathcal{C}_2$, $\mathcal{E} = \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\} \sqcup \forall r.\mathcal{E}_2$ and $\mathcal{C}_2 \Rightarrow_\alpha \mathcal{E}_2$.
  If $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\}) = \top$, we immediately obtain that $\mathrm{Supp}(\Upsilon, \mathcal{E}) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}) = \top$.
  We can now assume that $\mathrm{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\}) \neq \top$. It follows from the induction hypothesis that either
  (a) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) = \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{E}_2) = \top$, or
  (b) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$, or
  (c) $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$ and $\mathrm{Supp}(\Upsilon, \mathcal{C}_2) \Rightarrow^f_\alpha \mathcal{E}''$ with $\mathcal{E}'' \leq_s \mathrm{Supp}(\Upsilon, \mathcal{E}_2)$ and $\mathrm{Depth}(\mathcal{E}_2') \leq \mathrm{Depth}(\mathcal{E}_2)$.

If $\text{Supp}(\Upsilon, \mathcal{C}_2) = \top$, we obtain $\text{Supp}(\Upsilon, \mathcal{C}) = \top$ and $\text{Supp}(\Upsilon, \mathcal{E}_2) = \top$.

We now assume that $\text{Supp}(\Upsilon, \mathcal{C}_2) \neq \top$, i.e. either $\text{Supp}(\Upsilon, \mathcal{C}_2) \leq_s \text{Supp}(\Upsilon, \mathcal{E}_2)$ or $\text{Supp}(\Upsilon, \mathcal{C}_2) \Rightarrow^f_\alpha \mathcal{E}''$ with $\mathcal{E}'' \leq_s \text{Supp}(\Upsilon, \mathcal{E}_2)$ and $\text{Depth}(\mathcal{E}'_2) \leq \text{Depth}(\mathcal{E}_2)$ holds.

If $\text{Supp}(\Upsilon, \mathcal{C}_2) \leq_s \text{Supp}(\Upsilon, \mathcal{E}_2)$, we can infer that

$$\begin{aligned}\text{Supp}(\Upsilon, \mathcal{C}) &= \text{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\}) \sqcup \forall r.\text{Supp}(\Upsilon, \mathcal{C}_2) \\ &\leq_s \text{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\}) \sqcup \forall r.\text{Supp}(\Upsilon, \mathcal{E}_2) \\ &= \text{Supp}(\Upsilon, \mathcal{E})\end{aligned}$$

If $\text{Supp}(\Upsilon, \mathcal{C}_2) \Rightarrow^f_\alpha \mathcal{E}''$ with $\mathcal{E}'' \leq_s \text{Supp}(\Upsilon, \mathcal{E}_2)$ and $\text{Depth}(\mathcal{E}'_2) \leq \text{Depth}(\mathcal{E}_2)$, we can derive

$$\begin{aligned}\text{Supp}(\Upsilon, \mathcal{C}) \Rightarrow^f_\alpha &\text{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\}) \\ &\sqcup \forall r.\mathcal{E}'' \\ &\leq_s \text{Supp}(\Upsilon, \mathcal{C} \setminus \{\forall r.\mathcal{C}_2\}) \\ &\sqcup \forall r.\text{Supp}(\Upsilon, \mathcal{E}_2) \\ &= \text{Supp}(\Upsilon, \mathcal{E})\end{aligned}$$

□

Lemma 68. *Let $\mathcal{C}$ be a clause such that $\mathsf{sig}(\mathcal{C}) \cap \Upsilon \neq \emptyset$ and $\text{Supp}(\Upsilon, \mathcal{C}) = \top$. Additionally, let $\mathcal{D}, \mathcal{E}$ be clauses such that $\mathcal{C}(, \mathcal{D}) \Rightarrow_\alpha \mathcal{E}$ with $\alpha \in \mathsf{N_C} \setminus \Upsilon \cup \{\bot\}$.*

Then $\mathsf{sig}(\mathcal{E}) \cap \Upsilon \neq \emptyset$.

Lemma 69. *Let $\Upsilon \subseteq \mathsf{N_C}$ be a signature, let $\mathcal{M}$ be a set of initial clauses and let Δ be a $\mathfrak{T}$-derivation of a clause $\mathcal{E}$ from $\mathcal{M}$ such that $\text{Inferences}(\Delta) \cap \Upsilon = \emptyset$.*

Then either $\text{Supp}(\Upsilon, \mathcal{E}) = \top$ or there exists a $\mathfrak{T}^f$-derivation Δ' of a clause $\mathcal{E}'$ with $\mathcal{E}' \leq_s \text{Supp}(\Upsilon, \mathcal{E})$ and $\text{Inferences}(\Delta') \subseteq \text{Inferences}(\Delta) \cup \{\exists_f\}$ from $\text{Supp}(\Upsilon, \mathcal{M})$.

Proof. By induction on the depth d of Δ.

If $d = 0$, we can define the derivation Δ' to just consist of the clause $\mathcal{E}$, which then has the required properties.

Otherwise, $d > 0$ and we distinguish between the different inference rules that were used to derive the clause $\mathcal{E}$.

If the rule (i_2) was used to derive the clause $\mathcal{E}$, let $\mathcal{C}, \mathcal{D}$ be initial clauses such that $\mathcal{C}, \mathcal{D} \Rightarrow_\alpha \mathcal{E}$, $\alpha \in \mathsf{N_C} \setminus \Upsilon \cup \{\bot\}$ and let $\Delta_\mathcal{C}, \Delta_\mathcal{D}$ be the corresponding sub-derivations (of Δ) of the clauses $\mathcal{C}$ and $\mathcal{D}$, respectively. It follows from the induction hypothesis that for $x \in \{\mathcal{C}, \mathcal{D}\}$ either $\text{Supp}(\Upsilon, x) = \top$ or there exists a $\mathfrak{T}^f$-derivation $\Delta_{x'}$ of a clause x' with $x' \leq_s \text{Supp}(\Upsilon, x)$ and $\text{Inferences}(\Delta_{x'}) \subseteq \text{Inferences}(\Delta_x) \cup \{\exists_f\}$ from $\text{Supp}(\Upsilon, \mathcal{M})$. We obtain from Lemma 66 that either

(a) $\text{Supp}(\Upsilon, \mathcal{C}) = \top$ and $\text{Supp}(\Upsilon, \mathcal{E}) = \top$, or
(b) $\text{Supp}(\Upsilon, \mathcal{C}) = \top$ and $\text{Supp}(\Upsilon, \mathcal{D}) \leq_s \text{Supp}(\Upsilon, \mathcal{E})$, or
(c) $\text{Supp}(\Upsilon, \mathcal{D}) = \top$ and $\text{Supp}(\Upsilon, \mathcal{E}) = \top$, or
(d) $\text{Supp}(\Upsilon, \mathcal{D}) = \top$ and $\text{Supp}(\Upsilon, \mathcal{C}) \leq_s \text{Supp}(\Upsilon, \mathcal{E})$, or
(e) $\text{Supp}(\Upsilon, \mathcal{C}) \neq \top$, $\text{Supp}(\Upsilon, \mathcal{D}) \neq \top$ and $\text{Supp}(\Upsilon, \mathcal{C}), \text{Supp}(\Upsilon, \mathcal{D}) \Rightarrow^f_\alpha \mathcal{E}''$ with $\mathcal{E}'' \leq_s \text{Supp}(\Upsilon, \mathcal{E})$ and $\text{Depth}(\mathcal{E}') \leq \text{Depth}(\mathcal{E})$.

Hence, nothing remains to be shown in the cases (a) and (c). If (b) or (d) holds, we can define the derivation Δ' to consist of the derivation $\Delta_{\mathcal{D}'}$ or $\Delta_{\mathcal{C}'}$, respectively.

We can now assume that (e) holds. As $\mathcal{C}' \leq_s \text{Supp}(\Upsilon, \mathcal{C})$ and $\mathcal{D}' \leq_s \text{Supp}(\Upsilon, \mathcal{D})$ holds, we obtain from Lemma 64 that

- $\mathcal{C}' \leq_s \mathcal{E}''$, or
- $\mathcal{D}' \leq_s \mathcal{E}''$, or
- there exists a $\mathfrak{T}^f$-derivation Δ''' of a clause $\mathcal{E}'''$ from $\{\mathcal{C}', \mathcal{D}'\}$ such that
  – $\mathcal{E}''' \leq_s \mathcal{E}''$,
  – $\text{Inferences}(\Delta') \subseteq \{\alpha, \exists_f\}$,
  – $\mathcal{C}', \mathcal{D}' \in \text{Leaves}(\Delta')$.

In the first two cases we can define the derivation Δ' to consist of the derivation $\Delta_{\mathcal{C}'}$ or $\Delta_{\mathcal{D}'}$, respectively. If the third case above holds, we extend the derivations $\Delta_{\mathcal{C}'}$ and $\Delta_{\mathcal{D}'}$ to become a derivation of the clause $\mathcal{E}'''$.

Finally, the case where the rule (i_1) was used to derive the clause $\mathcal{E}$ can be proved analogously using Lemma 67. □

Theorem 70. *Let $\mathcal{T}$ be an $\mathcal{ALC}$ TBox, let $\Upsilon \subseteq \mathsf{N_C}$ be a signature, let $\succ \subseteq \Upsilon \times \Upsilon$ be a strict total order on Υ and let $m \in \mathbb{N}$. Then it holds that:*

(i) $\mathcal{T} \models \mathcal{F}_{\Sigma,m}(\mathcal{T})$

(ii) *For all* $\mathcal{ALC}$*-concepts* C, D *such that* $\mathsf{sig}(C,D) \subseteq \Sigma$ *and such that there exists a* $(\mathfrak{I}^u, \Upsilon, \succ)$*-refutation* Δ^u *from the universal clauses* $\mathrm{Cls}(\mathcal{T})$ *and the initial clauses* $\mathrm{Cls}(C \sqcap \neg D)$ *in which every clause is of depth at most* m*, it holds that*

$$\mathcal{F}_{\Sigma,m}(\mathcal{T}) \models C \sqsubseteq D.$$

Proof. (i) Easily follows from the properties of inference rules.

(ii) Let $\Upsilon = \mathsf{sig}(\mathcal{T}) \setminus \Sigma$,

$$\mathcal{S} = \mathrm{Res}^{\infty}_{\mathfrak{I}^u,\{A_n\},m}(\ldots \mathrm{Res}^{\infty}_{\mathfrak{I}^u,\{A_1\},m}(\mathcal{N})),$$

where $\succ$ is given by $A_1 \succ \ldots \succ A_n$, and let $\mathcal{F}_{\Sigma,m}(\mathcal{T}) = \mathrm{Supp}(\Upsilon, \mathcal{S})$. Additionally, let C, D be $\mathcal{ALC}$-concepts such that $\mathsf{sig}(\{C, D\}) \cap \Upsilon = \emptyset$ and such that there exists a $(\mathfrak{I}^u, \Upsilon)$-refutation Δ^u from the universal clauses $\mathrm{Cls}(\mathcal{T})$ and the initial clauses $\mathrm{Cls}(C \sqcap \neg D)$ in which every clause is of depth at most m.

Let $\mathcal{N} \subseteq \mathrm{Clauses}(\Delta^u)$ be all the clauses contained in the derivation Δ^u which have been obtained through Υ-inferences only. As $\mathsf{sig}(\{C, D\}) \subseteq \Sigma$, we can infer that $\mathcal{N}$ only contains universal clauses, i.e. $\mathcal{N} \subseteq \mathcal{S}$. It is then easy to see that there exists a $\mathfrak{I}$-refutation Δ from the initial clauses $\mathrm{Cls}(C \sqcap \neg D) \cup \{\, \forall \mathbf{r}.\mathcal{C} \mid \mathbf{r} \in \mathcal{R} \text{ and } \mathcal{C} \in \mathcal{R} \,\}$ such that $\mathrm{Inference}(\Delta) \cap \Upsilon = \emptyset$. Then, as $\mathrm{Supp}(\Upsilon, \bot) = \bot$, it follows from Lemma 69 that there exists a $\mathfrak{I}^f$-refutation Δ^f from

$$\begin{aligned}
&\mathrm{Supp}(\Upsilon, \mathrm{Cls}(C \sqcap \neg D)) \cup \mathrm{Supp}(\Upsilon, \{\, \forall \mathbf{r}.\mathcal{C} \mid \mathbf{r} \in \mathcal{R} \text{ and } \mathcal{C} \in \mathcal{R} \,\}) \\
&= \mathrm{Cls}(C \sqcap \neg D) \cup \{\, \forall \mathbf{r}.\mathrm{Supp}(\Upsilon, \mathcal{C}) \mid \mathbf{r} \in \mathcal{R} \text{ and } \mathcal{C} \in \mathcal{R} \,\} \\
&= \mathrm{Cls}(C \sqcap \neg D) \cup \{\, \forall \mathbf{r}.\mathcal{C} \mid \mathbf{r} \in \mathcal{R} \text{ and } \mathcal{C} \in \mathcal{F}_{\Sigma,m} \,\}
\end{aligned}$$

As $\mathfrak{I}^u$ is sound and by the fact that $\mathcal{F}_{\Sigma,m} \models \top \sqsubseteq \mathcal{D}$ for every $\mathcal{D} \in \{\, \forall \mathbf{r}.\mathcal{C} \mid \mathbf{r} \in \mathcal{R} \text{ and } \mathcal{C} \in \mathcal{F}_{\Sigma,m} \,\}$ we can conclude that $\mathcal{F}_{\Sigma,m} \models \bigsqcap_{\mathcal{E} \in \mathrm{Cls}(C \sqcap \neg D)} \mathcal{E} \sqsubseteq \bot$, i.e. $\mathcal{F}_{\Sigma,m} \models C \sqcap \neg D \sqsubseteq \bot$, which implies that $\mathcal{F}_{\Sigma,m} \models C \sqsubseteq D$. ☐

E General $\mathcal{ALC}$-TBoxes

We establish a bound on the length of the role sequence in the definition of an internalisation of a TBox based on the properties of model trees.

The *depth* of a model tree $\mathcal{M} = (V, E, L)$ is the maximal number of nodes obtained by an application of Operation 2 in any path in the tree $\mathcal{M}$.

Lemma 71. *Let* $\mathcal{T}$ *be an* $\mathcal{ALC}$*-TBox and let* E *be an* $\mathcal{ALC}$*-concept with* $\mathrm{Depth}(E) = n$*. Then it holds that*

$$\mathcal{T} \models \top \sqsubseteq E \quad \Longleftrightarrow \quad \models \bigsqcap_{(\mathbf{r},\mathcal{C}) \in \mathcal{P}_{\mathcal{T},n}} \forall \mathbf{r}.\mathcal{C} \sqsubseteq E$$

where

$$\begin{aligned}
\mathcal{P}_{\mathcal{T},n} = \{\, \forall \mathbf{r}.\mathcal{C} \mid\ & \mathcal{C} \in \mathrm{Cls}(\mathcal{T}), \mathbf{r} \in \mathcal{R}, \mathsf{sig}(r) \subseteq \mathsf{sig}(\mathcal{T}) \cup \mathsf{sig}(E), \\
& |\mathbf{r}| \le n + 2^{|\mathsf{sub}(\mathrm{Cls}(\mathcal{T}))|+1} \,\}
\end{aligned}$$

Proof. "$\Leftarrow$" follows immediately from the fact $\mathcal{T} \models \top \sqsubseteq \bigsqcap_{(\mathbf{r},\mathcal{C}) \in \mathcal{P}_{\mathcal{T},n}} \forall \mathbf{r}.\mathcal{C}$.

For the "$\Rightarrow$" direction, we show that (i) for any model tree $\mathcal{M} = (V, E, L)$ for $\mathrm{Cls}(\neg E)$ w.r.t. $\mathcal{T}$ of depth m it holds that $m \le n + 2^{|\mathsf{sub}(\mathrm{Cls}(\mathcal{T}))|+1}$, and (ii) there exists a model tree $\mathcal{M}' = (V', E', L')$ for $\mathcal{T}' = \emptyset$ and

$$\mathrm{Cls}(\bigsqcap_{(\mathbf{r},\mathcal{C}) \in \mathcal{P}_{\mathcal{T},n}} \forall \mathbf{r}.\mathcal{C} \sqcap \neg E)$$

such that there exists a bijective tree homomorphism $f \colon \mathcal{M} \to \mathcal{M}'$ with $L(x) = L'(f(x))$ for every $x \in V$.

Then, by the properties of homomorphism, the root node of $\mathcal{M}'$ is closed and so $(\bigsqcap_{(\mathbf{r},\mathcal{C}) \in \mathcal{P}_{\mathcal{T},n}} \forall \mathbf{r}.\mathcal{C} \sqcap \neg E)$ is unsatisfiable and thus $\models \bigsqcap_{(\mathbf{r},\mathcal{C}) \in \mathcal{P}_{\mathcal{T},n}} \forall \mathbf{r}.\mathcal{C} \sqsubseteq E$.

To prove (i) it suffices to notice that the label of any node in $\mathcal{M}$ to which there is a path in $\mathcal{M}$ containing n nodes obtained by an application of Operation 2, does not contain any clauses originating from $\mathrm{Cls}(E)$. From that moment, every node obtained by an application of Operation 2 only contains clauses contained in $\mathsf{sub}(\mathrm{Cls}(\mathcal{T}))$. As there are at most $2^{|\mathsf{sub}(\mathrm{Cls}(\mathcal{T}))|}$ of different sets of such subconcepts, in every path in $\mathcal{M}$ there can be at most $n + 2^{|\mathsf{sub}(\mathrm{Cls}(\mathcal{T}))|+1}$ nodes obtained by an application of Operation 2.

(ii) is proved by induction on the depth $m \le n + 2^{|\mathsf{sub}(\mathrm{Cls}(\mathcal{T}))|+1}$. For $m = 0$ the model tree $\mathcal{M}'$ coincides with the model tree $\mathcal{M}$. For $m > 0$ the induction step is proved by induction on the construction of $\mathcal{M}$.

☐

Proof of Theorem 11. It follows from Theorem 70 that $\mathcal{T} \models \mathcal{F}_{\Sigma,m}$ holds.

Now, let C and D be $\mathcal{ALC}$-concepts such that $\mathsf{sig}(C) \cup \mathsf{sig}(D) \subseteq \Sigma$, $\mathrm{Depth}(\{C, D\}) \leq n$ and $\mathcal{T} \models C \sqsubseteq D$, i.e. $\mathcal{T} \models \top \sqsubseteq \bigsqcap_{\mathcal{C}\in\mathrm{Cls}(\neg C\sqcup D)} \mathcal{C}$. By Lemma 71 we have that $\models \bigsqcap_{(\mathbf{r},\mathcal{C})\in\mathcal{P}_{\mathcal{T},n}} \forall\mathbf{r}.\mathcal{C} \sqsubseteq \neg C \sqcup D$, and thus, by Theorem 6 there exists a $(\mathfrak{I}, \Upsilon, \succ)$-derivation Δ of the empty clause from $\{\, \forall\mathbf{r}.\mathcal{C} \mid (\mathbf{r}, \mathcal{C}) \in \mathcal{P}_{\mathcal{T},n} \,\} \cup \mathrm{Cls}(C \sqcap \neg D)$ as

$$\mathrm{Cls}\left(\bigsqcap_{(\mathbf{r},\mathcal{C})\in\mathcal{P}_{\mathcal{T},n}} \forall\mathbf{r}.\mathcal{C}\right) = \{\, \forall\mathbf{r}.\mathcal{C} \mid (\mathbf{r}, \mathcal{C}) \in \mathcal{P}_{\mathcal{T},n} \,\}.$$

Note that for every clause $\mathcal{C} \in \mathrm{Clauses}(\Delta)$ it holds that $\mathrm{Depth}(\mathcal{C}) \leq m$. Consequently, by Theorem 7 there exists a $(\mathfrak{I}^u, \Upsilon, \succ)$-refutation Δ^u from the universal clauses $\mathrm{Cls}(\mathcal{T})$ and the initial clauses $\mathrm{Cls}(C \sqcap \neg D)$ such that $\mathrm{Depth}(\mathcal{C}) \leq m$ for every clause $\mathcal{C} \in \mathrm{Clauses}(\Delta^u)$. We can conclude that $\mathcal{F}_{\Sigma,m} \models C \sqsubseteq D$ holds by applying Theorem 70. □

F Acyclic $\mathcal{ALC}$-Terminologies

Definition 72. *Given an $\mathcal{ALC}$-terminology $\mathcal{T}$, we define a relation $\succ_{\mathcal{T}}$ over $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_C}$ as follows: for $A \bowtie C \in \mathcal{T}$, set $A \succ_{\mathcal{T}} B$ for every $B \in \mathsf{sig}(C) \cap \mathsf{N_C}$.*

An $\mathcal{ALC}$-terminology $\mathcal{T}$ is said to be acyclic *if, and only if, the relation $\succ_{\mathcal{T}}^{+}$ is irreflexive, where $\succ_{\mathcal{T}}^{+}$ denotes the transitive closure of the relation $\succ_{\mathcal{T}}$.*

Definition 73. *Let $\mathcal{T}$ be an acyclic $\mathcal{ALC}$-terminology. For $A \in \mathsf{N_C}$ we define the* definitorial depth *of A in $\mathcal{T}$ as follows:*

$$\mathit{DefinitorialDepth}_{\mathcal{T}}(A) = \max\{\, n \mid (A_0, \ldots, A_n) \in \mathsf{N_C}^n \text{ such that } A_i \succ_{\mathcal{T}} A_{i+1} \forall i : 0 \leq i \leq n-1 \,\}$$

Definition 74 (Unfolding of a concept w.r.t. $\mathcal{T}$)**.** *Let $\mathcal{T}$ be an acyclic $\mathcal{ALC}$-terminology, and let C be an $\mathcal{ALC}$-concept. Additionally, let $\mathcal{T} = \{A_1 \bowtie C'_1, \ldots, A_n \bowtie C'_n\}$ such that $\mathit{DefinitorialDepth}_{\mathcal{T}}(A_i) \geq \mathit{DefinitorialDepth}_{\mathcal{T}}(A_{i+1})$ for every $1 \leq i < n$. Then we define:*

$$C[\mathcal{T}] = (\ldots (C[A_1 \mapsto C'_1]) \ldots)[A_n \mapsto C'_n]$$

Lemma 75. *Let $\alpha = \{A \mapsto H\}$ be a substitution, let $\mathcal{I}$ be an interpretation and let $s \in \Delta^{\mathcal{I}}$.*

Then it holds for every $\mathcal{ALC}$-concept C that

$$s \in ([C]_\alpha)^{\mathcal{I}} \implies s \in (\neg(\bigsqcap_{\forall\mathbf{r}.G\in\mathcal{P}_{\{A\mapsto H\},C}} \forall\mathbf{r}.G) \sqcup C)^{\mathcal{I}}$$

where

$$\mathcal{P}_{\{A\mapsto H\},C} = \{\, \forall\mathbf{r}.((\neg A \sqcup H) \sqcap (\neg H \sqcup A)) \mid \mathbf{r} \in \mathcal{R} \text{ is a role-path to an occurrence of } A \text{ in } C \,\}$$

Proof. By induction on the structure of C.

If $C = A' \in \mathsf{N_C}$, the statement is obvious.

If $C = \exists r.D$, we have $s \in ([\exists r.D]_\alpha)^{\mathcal{I}} = (\exists r.[D]_\alpha)^{\mathcal{I}}$, i.e. there exists $t \in \Delta^{\mathcal{I}}$ with $(s, t) \in r^{\mathcal{I}}$ and $t \in ([D]_\alpha)^{\mathcal{I}}$. If we assume that $s \in (\bigsqcap_{\forall\mathbf{r}.G\in\mathcal{P}_{\{A\mapsto H\},C}} \forall\mathbf{r}.G)^{\mathcal{I}} = (\bigsqcap_{\forall r\mathbf{s}.G\in\mathcal{P}_{\{A\mapsto H\},D}} \forall r\mathbf{s}.G)^{\mathcal{I}}$, we can infer that $t \in (\bigsqcap_{\forall\mathbf{s}.G\in\mathcal{P}_{\{A\mapsto H\},D}} \forall\mathbf{s}.G)^{\mathcal{I}}$ and hence, $t \in D^{\mathcal{I}}$ by applying the induction hypothesis. We can conclude that $s \in C^{\mathcal{I}}$ holds.

The case for $C = \forall r.D$ is analogous to the previous case.

□

Lemma 76. *Let $\mathcal{T} = \{A_1 \equiv C_1, \ldots, A_n \equiv C_m\}$ be an acyclic $\mathcal{ALC}$-terminology such that $A_i \succeq_{\mathcal{T}} A_{i+1}$ for $1 \leq i \leq m$ and let C be an $\mathcal{ALC}$-concept. Additionally, let for a substitution $\alpha = \{A \mapsto H\}$ and for an $\mathcal{ALC}$-concept D, $F_\alpha[D]$ be the following concept*

$$F_\alpha[D] = \neg(\bigsqcap_{\forall\mathbf{r}.G\in\mathcal{P}_{\alpha,D}} \forall\mathbf{r}.G) \sqcup D$$

Finally, let $\mathcal{I}$ be an interpretation and let $s \in \Delta^{\mathcal{I}}$. Then it holds that

$$s \in ([C]_{\mathcal{T}})^{\mathcal{I}} \implies s \in (F_{\{A_1\mapsto C_1\}}[\ldots F_{\{A_{m-1}\mapsto C_{m-1}\}}[F_{\{A_m\mapsto C_m\}}[C]]])^{\mathcal{I}}$$

Proof. By induction on m using Lemma 75. □

Lemma 77. *Let $\mathcal{T} = \{A_1 \equiv C_1, \ldots, A_n \equiv C_m\}$ be an acyclic $\mathcal{ALC}$-terminology and let C be an $\mathcal{ALC}$-concept with $\mathrm{Depth}(C) = n$. Then it holds that*

$$\models \top \sqsubseteq [C]_{\mathcal{T}} \iff \models \bigsqcap_{\forall\mathbf{r}.G\in\mathcal{P}_{n,\mathcal{T}}} \forall\mathbf{r}.G \sqsubseteq C$$

where

$$\mathcal{P}_{n,\mathcal{T}} = \{\, \forall\mathbf{r}.((\neg A \sqcup F) \sqcap (\neg F \sqcup A)) \mid A \equiv F \in \mathcal{T}, \mathbf{r} \in \mathcal{R}, |\mathbf{r}| + \mathrm{Depth}(F) \leq n + \mathrm{ExpansionDepth}(\mathcal{T}) \,\}$$

Proof. "$\Rightarrow$" Assume $\models \top \sqsubseteq [C]_{\mathcal{T}}$. Let $\mathcal{I}$ be an interpretation with domain $\Delta^{\mathcal{I}}$ and let $s \in \Delta^{\mathcal{I}}$ such that $s \in (\bigsqcap_{\forall \mathbf{r}.G \in \mathcal{P}_{n,\mathcal{T}}} \forall \mathbf{r}.G)^{\mathcal{I}}$. It follows from the assumptions that $s \in ([C]_{\mathcal{T}})^{\mathcal{I}}$. Furthermore, for

$$F_{\{A_1 \mapsto C_1\}}[\ldots F_{\{A_{m-1} \mapsto C_{m-1}\}}[F_{\{A_m \mapsto C_m\}}[C]]] = \neg H_1 \sqcup (\ldots (\neg H_m \sqcup C)),$$

it is easy to see that

$$\models \bigsqcap_{\forall \mathbf{r}.G \in \mathcal{P}_{n,\mathcal{T}}} \forall \mathbf{r}.G \sqsubseteq H_1 \sqcap \ldots \sqcap H_m.$$

Hence, we can conclude that $s \in C^{\mathcal{I}}$ by applying Lemma 76.

"$\Leftarrow$" Assume $\models \bigsqcap_{\forall \mathbf{r}.G \in \mathcal{P}_{n,\mathcal{T}}} \forall \mathbf{r}.G \sqsubseteq C$ and let $\mathcal{I}$ be an interpretation. Now, let $\mathcal{I}'$ be an interpretation such that

- $\Delta^{\mathcal{I}'} = \Delta^{\mathcal{I}}$,
- $A^{\mathcal{I}'} = A^{\mathcal{I}}$ for every $A \in \mathsf{N_C} \setminus \{A_1, \ldots, A_n\}$,
- $r^{\mathcal{I}'} = r^{\mathcal{I}}$ for every $r \in \mathsf{N_R}$, and
- $A_i^{\mathcal{I}'} = \mathcal{C}_i^{\mathcal{I}'}$ for every $1 \leq i \leq n$.

Additionally, let $s \in \Delta^{\mathcal{I}} = \Delta^{\mathcal{I}'}$. Then, by definition of $\mathcal{I}'$ it is easy to see that $\mathcal{I}'$ is a model of $\mathcal{T}$ and that $s \in (\bigsqcap_{\forall \mathbf{r}.G \in \mathcal{P}_{n,\mathcal{T}}} \forall \mathbf{r}.G)^{\mathcal{I}'}$ holds, and thus, $s \in C^{\mathcal{I}'}$. Moreover, as $\mathcal{I}'$ is a model of $\mathcal{T}$, we have $C^{\mathcal{I}'} = ([C]_{\mathcal{T}})^{\mathcal{I}'}$. It remains to observe that $([C]_{\mathcal{T}})^{\mathcal{I}'} = ([C]_{\mathcal{T}})^{\mathcal{I}}$ as $\mathsf{sig}([C]_{\mathcal{T}}) \cap \{A_1, \ldots, A_n\} = \emptyset$, and we can conclude that $s \in ([C]_{\mathcal{T}})^{\mathcal{I}}$. ☐

Lemma 78. *Let $\mathcal{T}$ be an acyclic $\mathcal{ALC}$-terminology and let C be an $\mathcal{ALC}$-concept with* $\mathrm{Depth}(C) = n$. *Then it holds that*

$$\mathcal{T} \models \top \sqsubseteq C \quad \Longleftrightarrow \quad \models \bigsqcap_{\forall \mathbf{r}.\mathcal{C} \in \mathcal{P}_{\mathcal{T},n}} \forall \mathbf{r}.\mathcal{C} \sqsubseteq C$$

where

$$\mathcal{P}_{\mathcal{T},n} = \{\, \forall \mathbf{r}.\mathcal{C} \mid \mathcal{C} \in \mathrm{Cls}(\mathcal{T}), \mathbf{r} \in \mathcal{R}, \mathrm{Depth}(\forall \mathbf{r}.\mathcal{C}) \leq n + \mathrm{ExpansionDepth}(\mathcal{T}) \,\}$$

Proof. Let $\overline{\mathcal{T}} = \{\, A \equiv C \mid A \equiv C \in \mathcal{T} \,\} \cup \{\, A \equiv C \sqcap \bar{A} \mid A \sqsubseteq C \in \mathcal{T} \,\}$, where $\bar{A}$ are fresh concept names. Then the following equivalences hold:

$$\begin{aligned}
\mathcal{T} \models \top \sqsubseteq C \quad &\Longleftrightarrow \quad \overline{\mathcal{T}} \models \top \sqsubseteq C && \text{(see (Baader et al. 2007))} \\
&\Longleftrightarrow \quad \models \bigsqcap_{\forall \mathbf{r}.G \in \mathcal{P}_{n,\overline{\mathcal{T}}}} \forall \mathbf{r}.G \sqsubseteq C && \text{(Lemma 77)} \\
&\Longleftrightarrow \quad \models \bigsqcap_{\forall \mathbf{r}.G \in \mathcal{P}_{n,\mathrm{Cls}(\overline{\mathcal{T}})}} \forall \mathbf{r}.G \sqsubseteq C \\
&\Longleftrightarrow \quad \models \bigsqcap_{\forall \mathbf{r}.G \in \mathcal{P}_{n,\mathrm{Cls}(\mathcal{T})}} \forall \mathbf{r}.G \sqsubseteq C
\end{aligned}$$

(Note that the last equivalence follows from similar arguments as the first equivalence.) ☐

Proof of Corollary 12. It follows from Theorem 70 that $\mathcal{T} \models \mathcal{F}_{\Sigma,m}$ holds.

Now, let C, D be $\mathcal{ALC}$-concepts such that $\mathsf{sig}(C) \cup \mathsf{sig}(D) \subseteq \Sigma$, $\mathrm{Depth}(C, D) \leq n$, and $\mathcal{T} \models C \sqsubseteq D$. One can show analogously to the proof of Corollary 12 that $\mathcal{F}_{\Sigma,m}(\mathcal{T}) \models C \sqsubseteq D$ holds. ☐

G Correctness of the Implementation

In order to demonstrate the correctness of our implementation, we extend Lemma 64, which established the subsumption lemma for $\mathfrak{I}^f$, to $\mathfrak{I}^{u,f}$. Notice that Algorithm 1 only operates with universal clauses.

First we extend the minimal subsumption relation to universal clauses as follow: for two universal clauses $\mathcal{C}, \mathcal{D}$ we define

$$\mathcal{C} \leq_s^u \mathcal{D} \quad \Leftrightarrow \quad \exists \mathbf{r} \in \mathcal{R} \text{ such that } \forall \mathbf{r}.\mathcal{C} \leq_s \mathcal{D}.$$

Note that the relation $\leq_s^u$ is transitive.

Lemma 79. *Let $\mathcal{C}, \mathcal{D}, \mathcal{E}$ be universal clauses such that $\mathcal{C}, \mathcal{D} \vdash^{\alpha}_{\mathfrak{I}^{u,f}} \mathcal{E}$. Additionally, let $\mathcal{C}', \mathcal{D}'$ be universal clauses such that $\mathcal{C}' \leq_s^u \mathcal{C}$ and $\mathcal{D}' \leq_s^u \mathcal{D}$.*

Then one of the following propositions hold

- *$\mathcal{C}' \leq_s^u \mathcal{E}$, or*

- *$\mathcal{D}' \leq_s^u \mathcal{E}$, or*
- *there exists a $\mathfrak{T}^{u,f}$-derivation Δ' of a clause $\mathcal{E}'$ from $\{\mathcal{C}', \mathcal{D}'\}$ such that*
  - *$\mathcal{E}' \leq_s^u \mathcal{E}$,*
  - $\mathrm{Inferences}(\Delta') \subseteq \{\alpha, \exists_f\}$*, and*
  - $\mathcal{C}', \mathcal{D}' \in \mathrm{Leaves}(\Delta')$.

Proof. If the rule (mix) was used to derive the clause $\mathcal{E}$, let $\mathcal{C}, \forall \mathbf{r}.\mathcal{D} \Rightarrow_\alpha^f \mathcal{E}$. Moreover, let $\mathbf{r}_{\mathcal{C}'}, \mathbf{r}_{\mathcal{D}'} \in \mathcal{R}$ such that $\forall \mathbf{r}_{\mathcal{C}'}.\mathcal{C}' \leq_s \mathcal{C}$ and $\forall \mathbf{r}_{\mathcal{D}'}.\mathcal{D}' \leq_s \forall \mathbf{r}.\mathcal{D}$. It then follows from Lemma 64 that either

- $\forall \mathbf{r}_{\mathcal{C}'}.\mathcal{C}' \leq_s \mathcal{E}$, or
- $\forall \mathbf{r}_{\mathcal{D}'}.\mathcal{D}' \leq_s \mathcal{E}$, or
- there exists a $\mathfrak{T}^f$-derivation Δ' of a clause $\mathcal{E}''$ from $\{\forall \mathbf{r}_{\mathcal{C}'}.\mathcal{C}', \forall \mathbf{r}_{\mathcal{D}'}.\mathcal{D}'\}$ such that
  - $\mathcal{E}'' \leq_s \mathcal{E}$,
  - $\mathrm{Inferences}(\Delta') \subseteq \{\alpha, \exists_f\}$, and
  - $\forall \mathbf{r}_{\mathcal{C}'}.\mathcal{C}', \forall \mathbf{r}_{\mathcal{D}'}.\mathcal{D}' \in \mathrm{Leaves}(\Delta')$.

In the first two cases nothing remains to be shown as $\mathcal{C}' \leq_s^u \mathcal{C}$ or $\mathcal{D}' \leq_s^u \mathcal{D}$ holds, respectively. In the remaining case, it follows from Lemma 59 that there exists a $\mathfrak{T}^{u,f}$-derivation Δ'^u of a universal clause $\mathcal{E}'''$ from $\mathcal{N}$ such that

- $\mathcal{E}'' = \forall \mathbf{r}.\mathcal{E}'''$ where $\mathbf{r} = \mathrm{gcs}\{\mathbf{r}_{\mathcal{C}'}, \mathbf{r}_{\mathcal{D}'}\}$; and
- $\mathrm{Inferences}(\Delta'^u) \subseteq \{\alpha, \exists_f\}$, and
- $\mathcal{C}', \mathcal{D}' \in \mathrm{Leaves}(\Delta'^u)$.

By definition of the relation $\leq_s^u$, we can conclude that $\mathcal{E}''' \leq_s^u \mathcal{E}$ holds.

Finally, the cases where the rules (u_1) and (u_2) were used to derive the clause $\mathcal{E}$ can be proved analogously. $\square$