Journal of Universal Computer Science, vol. 16, no. 20 (2010), 3016-3042
submitted: 7/1/10, accepted: 13/9/10, appeared: 1/11/10

Providing a Proof-Theoretical Basis for Explanation: A Case Study on UML and $\mathcal{ALCQI}$ Reasoning

Alexandre Rademaker
(PUC-Rio, Rio de Janeiro, RJ, Brazil
`arademaker@inf.puc-rio.br`)

Edward Hermann Haeusler
(PUC-Rio, Rio de Janeiro, RJ, Brazil
`hermann@inf.puc-rio.br`)

Abstract: In this article we argue in favour of Natural Deduction Systems as a basis for formal proof explanations. We illustrate our choice presenting a Natural Deduction for $\mathcal{ALCQI}$ and use it to help explain UML reasoning.
Key Words: ALC, Description Logics, UML, ALCQI, Proof Theory, Sequent Calculus, Natural Deduction
Category: F.4.1, M.4

1 Introduction

Description Logics (DL) are quite well-established as underlying logics for KR. $\mathcal{ALC}$ [1] is a basic description logic. UML is among the most used semi-formal artifacts in computer science. The DL-community has shown that one needs to go a bit further to reason on UML models. $\mathcal{ALCQI}$ [2] is able to express most of the features involved in an UML modeling. DL-Lite could also be taken for this, although it might be more verbose.

When we define a theory, from UML models, the reasoner should provide understandable explanations in order to facilitate the process of evolving the theory towards its validation or extension. There are some works on explanation in DL, we cite [McGuinness, 1996, Calvanese et al., 2004, Borgida et al., 1999, Liebig & Halfmann, 2005] among them. They rely on the proof system implemented by the reasoner. Tableaux and Sequent Calculus (SC) are the main proof systems used. On the other hand, Natural Deduction (ND) is a proof system that tries to naturally represent human mathematical/formal reasoning, at least Gentzen[3] aimed at this. Prawitz improved it, on top of Jaskowski's work, characterizing proofs without detours, called *Normal Proofs.* They correspond

[1] Attributive Concept Language with Complements.
[2] The extension of $\mathcal{ALC}$ with quantified number restrictions ($\mathcal{Q}$) and inverse of roles constructor ($\mathcal{I}$).
[3] "First I wished to construct a formalism that comes as close as possible to actual reasoning. Thus arose a *calculus of natural deduction*" [Gentzen, 1935].

to Analytic Tableaux and cut-free SC proofs. Curry observed that the procedure that yields normal proofs from non-normal ones is related to the way typed λ-terms are evaluated. This is, nowadays, known as the Curry-Howard isomorphism between algorithms and ND proofs. The typed λ-term associated to a ND deduction is taken as its computational content. We believe that the computational content of ND helps in choosing it as the basis to generate adequate explanations on theorem hood in a theory.

We discuss why ND is the most adequate structure to explain theorems and then use a ND for $\mathcal{ALCQI}$ to explain reasoning on an UML model cited by the DL-community. A natural deduction calculus for the core logic $\mathcal{ALC}$ is shown and then extended to $\mathcal{ALCQI}$. In the following we discuss ND, Analytic Tableaux and Sequent Calculus as a basis to explanation generation. In section 6 we compare the use of these systems in providing explanation on UML reasoning. We mention that according [Berardi et al., 2005], $\mathcal{ALCQI}$ is enough and adequate to express UML class diagram consistency. Our work is strongly based on this results.

Finally, this article is an extended revision of [Haeusler & Rademaker, 2009] based on reviewer's suggestions. The main modifications are: (1) the proofs presented in sections 3 and 4 are now fully presented, we revised other proofs and improve explanations about each proof step; (2) the notation of labeled concepts in section 3 and 5 were completely redefined and simplified. The authors would like to thank the reviewers insightful comments.

2 Proofs and Explanations

From a logical point of view, conceptual modeling tasks can be summarized by the following steps:

1. Observe the "World";
2. Determine what is relevant;
3. Choose or define your terminology by means of non-logical linguistic terms;
4. Write down the main laws, the axioms, governing your "World";
5. Verify the correctness (sometimes completeness too) of your set of laws, that is, the theory constructed.

Steps 1, 2 and 3 may be facilitated by the use of an informal notation (UML, ER, Flow-Charts, etc) and their respective methodology, but it is essentially "Black Art" [Maibaum, 2005]. Step 4 demands quite a lot of knowledge of the "world" begin specified (the model). Step 5 essentially provides finitely many tests as support for the correctness of an infinitely quantified property.

A deduction of a proposition α from a set of hypotheses Γ is essentially a means of providing convincing evidence that Γ entails α. When validating a theory, represented by a set of logical formulae, we mainly test entailments, possibly using a theorem prover. Considering a model M specified by the set of axioms $Spec(M)$, given a property ϕ about M, from the entailment tests results one can rise the following questions:

1. If $M \models \phi$ and $Spec(M) \vdash \phi$, why ϕ is truth? One must provide a proof of ϕ;
2. If $M \models \phi$, but $Spec(M) \nvdash \phi$ from the attempt to construct the proof of ϕ one may obtain a counter-model and from that counter-model an explanation for the failed entailment. Model-checking based reasoning can be used in such situation;
3. If $M \not\models \phi$, but $Spec(M) \vdash \phi$, why does this false proposition holds? In this case, one must provide a proof of ϕ.

Here we are interested in the last case, tests providing a false positive answer, that is, the prover shows a deduction/proof for an assertion that must be invalid in the considered theory. This is one of the main reasons to explain a theorem when validating a theory, provide explanation of why a false positive is entailed. Another reason to provide explanations on a theorem, has to do with providing explanation of why some assertion is a true positive, the first case. This latter use is concerned to certification, in this case the proof/deduction itself serves as a certification document. This article does not take into account educational uses of theorem provers, and their resulting theorems, since explanations in these cases are more demanding.

For the tasks of providing proofs and explanations, we compare three deduction systems, Analytic Tableaux (**AT**) [Smullyan, 1968], Sequent Calculus (**SC**) [Takeuti, 1975] and Natural Deduction (**ND**) [Prawitz, 1965] as presented in the respective cited references. Because of the lack of space we do not show the set of rules for each system. Nevertheless, they are quite well-known and this may not prejudice reader's understanding. In this section we consider the propositional logic as defined in [Prawitz, 1965]). Let us consider a theory (presented by a knowledge base $\mathcal{KB}$) containing the single axiom

$$\mathcal{KB} \equiv (Quad \wedge PissOnFireHydrant) \rightarrow Dog$$

which classifies a *dog* as a *quadruped* that pisses on a fire hydrant. This $\mathcal{KB}$ draws the following proposition

$$(Quad \rightarrow Dog) \vee (PissOnFireHydrant \rightarrow Dog)$$

Figure 1 presents three from many more possible proofs of this entailment in the Tableaux system. In the proofs of Figure 1, the symbols $\mathscr{V}$ and $\mathscr{F}$ stands for “verum” and “falsum”, respectively, we adopted them instead of “true” and “false” as a matter of style. Figure 2 presents three possible proofs in Sequent Calculus, they are also sorted out from many others possible proofs in Sequent Calculus. Figure 3 presents the only two possible normal proofs for this entailment.

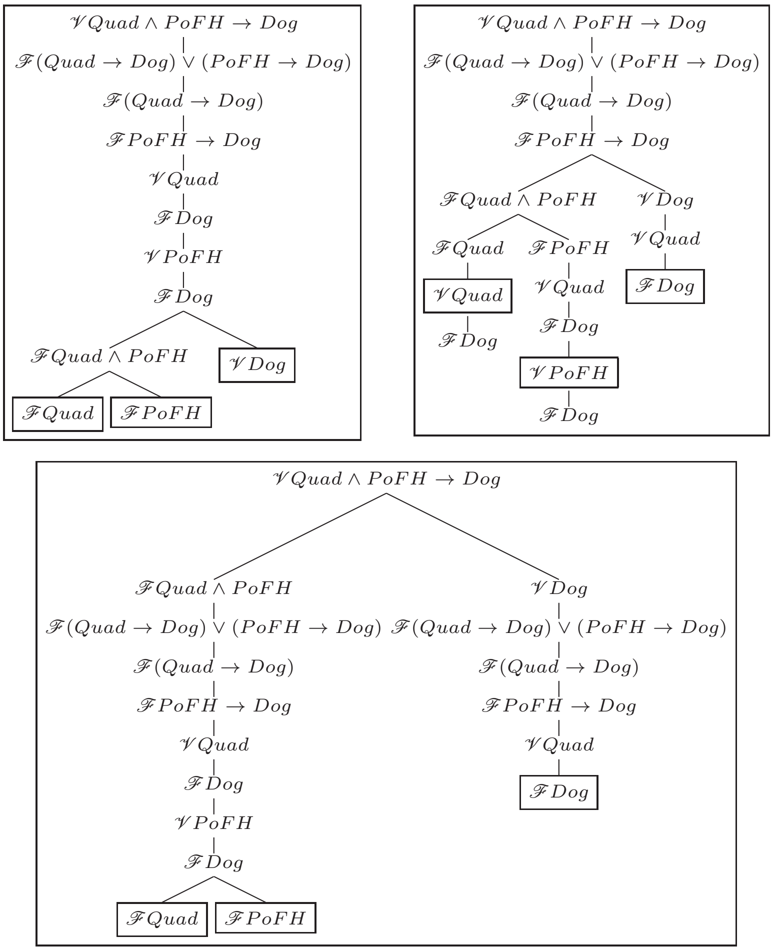

Figure 1: Tableaux proofs

Consider the derivations from Figure 1 and 2. They correspond to the Natural Deduction derivations that are shown in Figure 3. The Tableaux and Sequent Calculus variants only differ in the order of rule applications. In **ND** there is no such distinction. In this example, the order of application is irrelevant in terms

$$\dfrac{\dfrac{\dfrac{\mathcal{KB} \Rightarrow PoFH \wedge Quad \to Dog \quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{Quad \Rightarrow Quad}{Quad, PoFH \Rightarrow Quad} \quad \dfrac{PoFH \Rightarrow PoFH}{Quad, PoFH \Rightarrow PoFH}}{Quad, PoFH \Rightarrow Quad \wedge PoFH} \quad Dog \Rightarrow Dog}{Quad, PoFH, PoFH \wedge Quad \to Dog \Rightarrow Dog}}{Quad, PoFH, PoFH \wedge Quad \to Dog \Rightarrow Dog, Dog}}{PoFH, PoFH \wedge Quad \to Dog \Rightarrow (Quad \to Dog), Dog}}{PoFH, \mathcal{KB} \Rightarrow (Quad \to Dog), Dog}}{\mathcal{KB} \Rightarrow (Quad \to Dog), (PoFH \to Dog)}}{\mathcal{KB} \Rightarrow (Quad \to Dog) \vee (PoFH \to Dog)}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{KB} \Rightarrow PoFH \wedge Quad \to Dog \quad \dfrac{\dfrac{\dfrac{Quad \Rightarrow Quad}{Quad, PoFH \Rightarrow Quad} \quad \dfrac{PoFH \Rightarrow PoFH}{Quad, PoFH \Rightarrow PoFH}}{Quad, PoFH \Rightarrow Quad \wedge PoFH} \quad Dog \Rightarrow Dog}{Quad, PoFH, PoFH \wedge Quad \to Dog \Rightarrow Dog}}{\mathcal{KB}, Quad, PoFH \Rightarrow Dog}}{\mathcal{KB}, Quad, PoFH \Rightarrow Dog, Dog}}{\mathcal{KB}, PoFH \Rightarrow (Quad \to Dog), Dog}}{\mathcal{KB} \Rightarrow (Quad \to Dog), (PoFH \to Dog)}}{\mathcal{KB} \Rightarrow (Quad \to Dog) \vee (PoFH \to Dog)}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{KB} \Rightarrow PoFH \wedge Quad \to Dog \quad \dfrac{\dfrac{\dfrac{Quad \Rightarrow Quad}{Quad, PoFH \Rightarrow Quad} \quad \dfrac{PoFH \Rightarrow PoFH}{Quad, PoFH \Rightarrow PoFH}}{Quad, PoFH \Rightarrow Quad \wedge PoFH} \quad \dfrac{Dog \Rightarrow Dog}{Dog \Rightarrow Dog, Dog}}{Quad, PoFH, PoFH \wedge Quad \to Dog \Rightarrow Dog, Dog}}{\mathcal{KB}, Quad, PoFH \Rightarrow Dog, Dog}}{\mathcal{KB}, PoFH \Rightarrow (Quad \to Dog), Dog}}{\mathcal{KB}, PoFH \Rightarrow (Quad \to Dog), Dog}}{\dfrac{\mathcal{KB} \Rightarrow (Quad \to Dog), (PoFH \to Dog)}{\mathcal{KB} \Rightarrow (Quad \to Dog) \vee (PoFH \to Dog)}}$$

Figure 2: Sequent Calculus proofs

of explanation, although it is not for the prover's implementation. The pattern represented by the **ND** deduction is close to what one expects from an argument drawing a conclusion from any conjunction that it contains. This example shows how **SC** proofs carry more information than the need for a meaningful explanation. Concerning the **AT** system, Smullyan noted that its proofs correspond to **SC** proofs by considering sequents formed by positively signed formulas ($T\alpha$) at the antecedent and negatively signed ones ($F\alpha$) appearing at the consequent. A Block **AT** is defined then by considering **AT** expansion rules in the form of inference rules. In this way, our example in **SC** would carry the same content useful for explanation carried by the **AT** proofs. We must note that the different **SC** proofs and its corresponding **AT** proofs, as the ones shown, are represented, all of them, by only two possible variations of normal derivations in **ND**.

Natural Deduction seems to be the oldest among the three systems here considered. Gentzen decided to move from **ND** to **SC** in order to detour from technical problems faced by him in his syntactical proof of the consistency of Arithmetic in 1936. As mentioned by [Prawitz, 1965], **SC** can be understood as

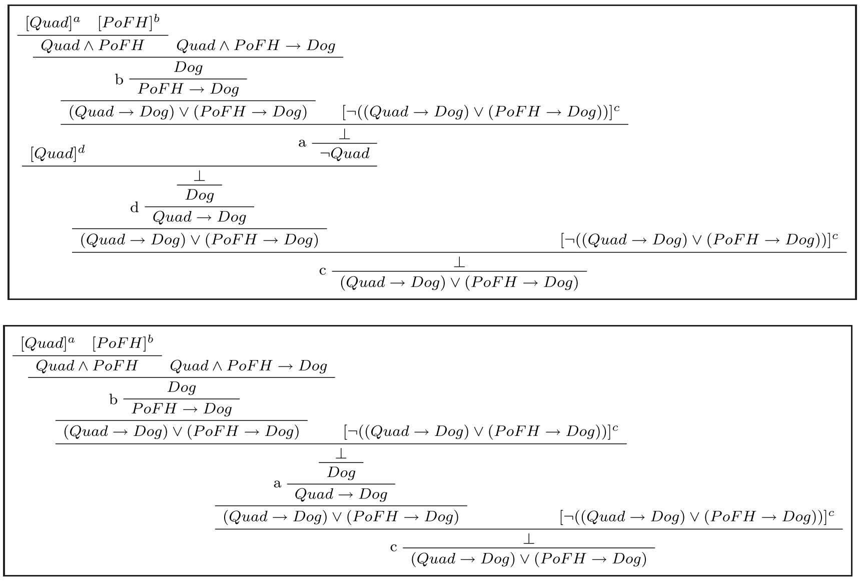

Figure 3: Natural Deduction proofs

a meta-calculus for the deducible relation in **ND**. A consequence of this is that **ND** can represent in only one deduction of α from $\gamma_1, \ldots, \gamma_n$ many **SC** proofs of the sequent $\gamma_1, \ldots, \gamma_n \Rightarrow \alpha$. Gentzen made **SC** formally state rules that were implicit in **ND**, such as the structural rules. We advice the reader that the **SC** used here (see [Takeuti, 1975]) is a variation of Gentzen's calculus designed with the goal of having, in each inference rule, any formula occurring in a premise as a sub-formula of some formula occurring in the conclusion. This sub-formula property facilitates the implementation of a prover based on this very system.

Consider a normal **ND** deduction Π_1 of α from $\{\gamma_1, \ldots, \gamma_k\}$, and, a deduction Π_2 of γ_i (for some $i = 1, \ldots, k$) from $\{\delta_1, \ldots, \delta_n\}$. Using latter Π_1 in the former Π_2 deduction yields a (possibly non-normal) deduction of α from $\{\gamma_1, \ldots, \gamma_k, \delta_1, \ldots, \delta_n\}$. This can be done in **SC** by applying a cut rule between the proofs of the corresponding sequents $\delta_1, \ldots, \delta_n \Rightarrow \gamma_i$ and $\gamma_1, \ldots, \gamma_k \Rightarrow \alpha$ yielding a proof of the sequent $\gamma_1, \ldots, \gamma_k, \delta_1, \ldots, \delta_n \Rightarrow \alpha$. The new **ND** deduction can be normalized, in the former case, and the cut introduced in the latter case can be eliminated. In the case of **AT**, the fact that they are closed by *modus ponens* implies that closed **AT** for $\delta \to \gamma$ and $\gamma \to \alpha$ entails the existence of a closed **AT** for $\delta \to \alpha$. The use of cuts, or equivalently, lemmas may reduce the size of a derivation. However, the relevant information conveyed by a deduction or proof in any of these systems has to firstly consider normal deductions, cut-free proofs and analytic Tableaux. They are the most representative formal objects in each of these systems as a consequence of the sub-formula property, holding in **ND** too. Besides that they are computationally easier to build than their non-normal counterparts.

These examples are carried out in Minimal Logic. [4] For Classical reasoning, an inherent feature of most DLs, including $\mathcal{ALC}$, the above scenario changes. Any classical proof of the sequent $\gamma_1, \gamma_2 \Rightarrow \alpha_1, \alpha_2$ corresponds a **ND** deduction of $\alpha_1 \vee \alpha_2$ from γ_1, γ_2, or, of α_1 from $\gamma_1, \gamma_2, \neg\alpha_2$, or, of α_2 from $\gamma_1, \gamma_2, \neg\alpha_1$, or, of $\neg\gamma_1$ from $\neg\alpha_1, \gamma_2, \neg\alpha_2$, and so on. In Classical logic [5], each **SC** may represent more than one deduction, since we have to choose which formula will be the conclusion in the **ND** side. We recall that it still holds that to each **ND** deduction there is more than one **SC** proof. In order to serve as a good basis for explanations of classical theorems we choose **ND** as the most adequate. Note that we are not advocating that the prover has to produce **ND** proofs directly. An effective translation to a **ND** might be provided. Of course there must be a **ND** for the logic involved. If, besides that, a normalization is provided for a system, we know that it is possible to always deal with canonical proofs satisfying the sub-formula principle. In the following we present a **ND** for $\mathcal{ALC}$.

In section 6 an example illustrating the use of proofs to explain reasoning on UML models is accomplished in **ND**, **SC** and **AT**.

3 A Natural Deduction for $\mathcal{ALC}$

In this section we present a Natural Deduction (ND) system for $\mathcal{ALC}$, named $\text{ND}_{\mathcal{ALC}}$. We briefly discuss the motivation and the basic considerations behind the design of $\text{ND}_{\mathcal{ALC}}$. We also show the completeness, soundness and normalization theorems.

$\mathcal{ALC}$ is a basic description language [Baader et al., 2003] and its syntax of concept descriptions, denoted as C, is described as follows:

$$C ::= \bot \mid \top \mid A \mid \neg C \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \exists R.C \mid \forall R.C$$

where A stands for atomic concepts and R for atomic roles. The concept $\top$ can be taken as $\neg\bot$.

The semantics of concept descriptions is defined in terms of an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$. The domain $\Delta^{\mathcal{I}}$ of $\mathcal{I}$ is a non-empty set of individuals and the interpretation function $\cdot^{\mathcal{I}}$ maps each atomic concept A to a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and for each atomic role r a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The interpretation function $\cdot^{\mathcal{I}}$ is extended to concept descriptions inductively as follows:

[4] Minimal logic is obtained from Intuitionistic logic removing the "ex-falso quodlibet" rule, that is, the rule that allows to deduce anything from a logical contradiction.

[5] Intuitionistic Logic and Minimal Logic have similar behavior concerning the relationship between their respective systems of **ND** and **SC**.

$$
\begin{aligned}
\top^{\mathcal{I}} &= \Delta^{\mathcal{I}} \\
\bot^{\mathcal{I}} &= \emptyset \\
(\neg C)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \\
(C \sqcap D)^{\mathcal{I}} &= C^{\mathcal{I}} \cap D^{\mathcal{I}} \\
(C \sqcup D)^{\mathcal{I}} &= C^{\mathcal{I}} \cup D^{\mathcal{I}} \\
(\exists R.C)^{\mathcal{I}} &= \{a \in \Delta^{\mathcal{I}} \mid \exists b.(a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\} \\
(\forall R.C)^{\mathcal{I}} &= \{a \in \Delta^{\mathcal{I}} \mid \forall b.(a,b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\}
\end{aligned}
$$

The $\text{ND}_{\mathcal{ALC}}$ presented in Figure 4 is based on the extension of the $\mathcal{ALC}$ language in which concepts are decorated by a list of labels. Its syntax is as following:

$$
\begin{aligned}
LB &\rightarrow \forall R \mid \exists R \\
L &\rightarrow LB, L \mid \emptyset \\
\phi_{lc} &\rightarrow {}^{L}\phi_c
\end{aligned}
$$

where R stands for atomic role names, LB for a label and L for list of labels and ϕ_c for $\mathcal{ALC}$ concept descriptions. That is, labels are nothing but existencial or universal quantified roles names. We say that a labeled $\mathcal{ALC}$ concept is *consistent* if it has an $\mathcal{ALC}$ concept equivalent. For instance, if $\exists R_2.\forall Q_2.\exists R_1.\forall Q_1.\alpha$ is an $\mathcal{ALC}$ concept, ${}^{\exists R_2.\forall Q_2.\exists R_1.\forall Q_1}\alpha$ is its labeled concept equivalent. [6] Labels are syntactic artifacts of our system, which means that labeled concepts and its equivalent $\mathcal{ALC}$ have the same semantics. $\text{ND}_{\mathcal{ALC}}$ was designed to be extended to DLs with role constructors and subsumptions. This is one of the main reasons to use roles-as-labels in its formulation.

The notation ${}^{\neg L}\alpha$ denotes the exchanging of the universal and existential roles occurring in L in a consistent way. This is used to express the negation of labeled concepts. That is, if $\gamma \equiv {}^{\exists R_2.\forall Q_2.\exists R_1.\forall Q_1}\alpha$, then we can express its negation by $\neg\gamma \equiv {}^{\forall R_2.\exists Q_2.\forall R_1.\exists Q_1}\neg\alpha$.

In the rule $\sqsubseteq$-i, ${}^{L_1}\alpha \sqsubseteq {}^{L_2}\beta$ depends only on the assumption ${}^{L_1}\alpha$ and no other hypothesis. The proviso to rule *Gen* application is that the premise ${}^{L}\alpha$ does not depend on any hypothesis. In $\bot_c$-rule, ${}^{L}\alpha$ has to be different from $\bot$. In some rules the list of labels L has a superscript, $L^{\forall}$ or $L^{\exists}$. This notation means that the list of labels L should contain only $\forall R$ (resp. $\exists R$) labels. When L has no superscript, any kind of label is allowed.

Despite the use of labelled formulas, the main non-standard feature of $\text{ND}_{\mathcal{ALC}}$ is the fact that it is defined on two kinds of objects, namely *concept descriptions*

[6] One can easly define a function $\sigma : \phi_{lc} \rightarrow \phi_c$ to transform labelled concept into its equivalent $\mathcal{ALC}$ concept. That is, for example $\sigma({}^{\exists R_2.\forall Q_2.\exists R_1.\forall Q_1}\alpha) = \exists R_2.\forall Q_2.\exists R_1.\forall Q_1.\alpha$.

$$\frac{{}^{L^\forall}(\alpha \sqcap \beta)}{{}^{L^\forall}\alpha}\ \sqcap\text{-e} \qquad \frac{{}^{L^\forall}(\alpha \sqcap \beta)}{{}^{L^\forall}\beta}\ \sqcap\text{-e} \qquad \frac{{}^{L^\forall}\alpha \quad {}^{L^\forall}\beta}{{}^{L^\forall}(\alpha \sqcap \beta)}\ \sqcap\text{-i}$$

$$\frac{{}^{L^\exists}(\alpha \sqcup \beta) \quad \begin{matrix}[{}^{L^\exists}\alpha]\\ \vdots \\ \gamma\end{matrix} \quad \begin{matrix}[{}^{L^\exists}\beta]\\ \vdots \\ \gamma\end{matrix}}{\gamma}\ \sqcup\text{-e} \qquad \frac{{}^{L^\exists}\alpha}{{}^{L^\exists}(\alpha \sqcup \beta)}\ \sqcup\text{-i} \qquad \frac{{}^{L^\exists}\beta}{{}^{L^\exists}(\alpha \sqcup \beta)}\ \sqcup\text{-i}$$

$$\frac{{}^{L}\alpha \quad {}^{\neg L}\neg\alpha}{\bot}\ \neg\text{-e} \qquad \frac{\begin{matrix}[{}^{L}\alpha]\\ \vdots \\ \bot\end{matrix}}{{}^{\neg L}\neg\alpha}\ \neg\text{-i} \qquad \frac{\begin{matrix}[{}^{\neg L}\neg\alpha]\\ \vdots \\ \bot\end{matrix}}{{}^{L}\alpha}\ \bot_c$$

$$\frac{{}^{L}\exists R.\alpha}{{}^{L,\exists R}\alpha}\ \exists\text{-e} \qquad \frac{{}^{L,\exists R}\alpha}{{}^{L}\exists R.\alpha}\ \exists\text{-i} \qquad \frac{{}^{L}\forall R.\alpha}{{}^{L,\forall R}\alpha}\ \forall\text{-e}$$

$$\frac{{}^{L,\forall R}\alpha}{{}^{L}\forall R.\alpha}\ \forall\text{-i} \qquad \frac{{}^{L}\alpha}{{}^{\forall R,L}\alpha}\ Gen$$

$$\frac{{}^{L_1}\alpha \quad {}^{L_1}\alpha \sqsubseteq {}^{L_2}\beta}{{}^{L_2}\beta}\ \sqsubseteq\text{-e} \qquad \frac{\begin{matrix}[{}^{L_1}\alpha]\\ \vdots \\ {}^{L_2}\beta\end{matrix}}{{}^{L_1}\alpha \sqsubseteq {}^{L_2}\beta}\ \sqsubseteq\text{-i}$$

Figure 4: The Natural Deduction system for $\mathcal{ALC}$

and *subsumptions*. Concept descriptions are interpreted as sets. [7] On the other hand, a subsumption $\alpha \sqsubseteq \beta$, with α and β being concepts, is a truth-value statement. Its truth depends on whether the interpretation of β includes the corresponding interpretation of α.

The semantics of $\mathrm{ND}_{\mathcal{ALC}}$ follows the $\mathcal{ALC}$ semantics, that is, it is given by an *interpretation*. However, since $\mathrm{ND}_{\mathcal{ALC}}$ deals with two different kinds of objects, we must define how an interpretation satisfies both kinds.

Definition 1. Let $\Omega = (\mathcal{C}, \mathcal{S})$ be a tuple composed by a set of labeled concepts $\mathcal{C} = \{\alpha_1, \ldots, \alpha_n\}$ and a set of subsumption $\mathcal{S} = \{\gamma_1 \sqsubseteq \delta_1, \ldots, \gamma_k \sqsubseteq \delta_k\}$. We say that an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ satisfies Ω and write $\mathcal{I} \models \Omega$ whenever: (i)

[7] See the presented $\mathcal{ALC}$ semantics.

$\mathcal{I} \models \mathcal{C}$, which means $\bigcap_{\alpha \in \mathcal{C}} (\alpha)^{\mathcal{I}} \neq \emptyset$; and (ii) $\mathcal{I} \models \mathcal{S}$, which means that for all $(\gamma_i \sqsubseteq \delta_i) \in \mathcal{S}$, we have $(\gamma_i)^{\mathcal{I}} \subseteq (\delta_i)^{\mathcal{I}}$.

We adopted the standard notation $\Omega \vdash F$ if there exists a deduction Π with conclusion F (concept or subsumption) from Ω as set of hypothesis.

From [Schild, 1991] we known that $\mathcal{ALC}$ is sound and complete for any Classical Propositional Logic axiomatization containing the axioms:

Definition 2 An Axiomatization of $\mathcal{ALC}$.

$$\forall R.(\alpha \sqcap \beta) \equiv \forall R.\alpha \sqcap \forall R.\beta \tag{1}$$

$$\forall R.\top \equiv \top \tag{2}$$

As usual, $\exists R.\alpha$ can be taken as a shorthand for $\neg\forall R.\neg\alpha$, as well as $\forall R.\alpha$ as a shorthand for $\neg\exists R.\neg\alpha$. Taking $\exists R.\alpha$ as the *definienens* concept, the Axiom 1 can be rewritten to Axiom 3.

$$\exists R.(\alpha \sqcup \beta) \equiv \exists R.\alpha \sqcup \exists R.\beta \tag{3}$$

The following rule, also known as necessitation rule:

$$\frac{\vdash \alpha}{\vdash \forall R.\alpha}\ Nec$$

is sound and complete for $\mathcal{ALC}$ semantics. In fact, the Axiom 1 and this necessitation rule are an alternative axiomatization for $\mathcal{ALC}$.

In what follows, we proof that $\text{ND}_{\mathcal{ALC}}$ is sound and complete.

Theorem 3. $\text{ND}_{\mathcal{ALC}}$ *is complete regarding the standard semantics of* $\mathcal{ALC}$.

Proof. To prove Theorem 3 we show how the axiomatic presentation of $\mathcal{ALC}$ from [Schild, 1991] can be derived in $\text{ND}_{\mathcal{ALC}}$.

The $\mathcal{ALC}$ necessitation rule (Nec) is a derived rule of $\text{ND}_{\mathcal{ALC}}$, for supposing $\vdash \alpha$ implies the existence of a proof (without hypothesis) Π of α. We prove $\forall R.\alpha$, without any new hypothesis by means of the following schema:

$$\cfrac{\cfrac{\begin{array}{c}\Pi \\ \vdots \\ \alpha\end{array}}{{}^{R}\alpha}\ Nec}{\forall R.\alpha}\ \forall\text{-i}$$

The following proofs justify in $\text{ND}_{\mathcal{ALC}}$ the $\mathcal{ALC}$ axiom

$$\forall R.(A \sqcap B) \equiv (\forall R.A \sqcap \forall R.B)$$

where $\alpha \equiv \beta$ is an abbreviation for $\alpha \sqsubseteq \beta$ and $\beta \sqsubseteq \alpha$, having obvious $\equiv$ elimination and introduction rules, based on $\sqsubseteq$ elimination and introduction rules.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\forall R.(A \sqcap B)]}{{}^{R}(A \sqcap B)}\,\forall\text{-e}}{{}^{R}A}\,\sqcap\text{-e}}{\forall R.A}\,\forall\text{-i} \qquad \dfrac{\dfrac{\dfrac{[\forall R.(A \sqcap B)]}{{}^{R}(A \sqcap B)}\,\forall\text{-e}}{{}^{R}B}\,\sqcap\text{-e}}{\forall R.B}\,\forall\text{-i}}{\forall R.A \sqcap \forall R.B}\,\sqcap\text{-i}}{\forall R.(A \sqcap B) \sqsubseteq \forall R.A \sqcap \forall R.B}\,\sqsubseteq\text{-i}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\forall R.A \sqcap \forall R.B]}{\forall R.A}\,\sqcap\text{-e}}{{}^{R}A}\,\forall\text{-e} \qquad \dfrac{\dfrac{[\forall R.A \sqcap \forall R.B]}{\forall R.B}\,\sqcap\text{-e}}{{}^{R}B}\,\forall\text{-e}}{{}^{R}(A \sqcap B)}\,\sqcap\text{-i}}{\forall R.(A \sqcap B)}\,\forall\text{-i}}{\forall R.A \sqcap \forall R.B \sqsubseteq \forall R.(A \sqcap B)}\,\sqsubseteq\text{-i}$$

$\mathrm{ND}_{\mathcal{ALC}}$ is a conservative extension of the classical propositional calculus. To see that, let Δ be a set of formulas of the form $\{\gamma_1, \ldots, \gamma_k, \alpha_1 \rightarrow \beta_1, \ldots, \alpha_n \rightarrow \beta_n\}$, where each γ_i, α_i and β_i are propositional formulas and α_i and β_i do not have any occurrence of $\rightarrow$. One can easily verify that any propositional classical consequence $\Delta \models \alpha$ is justified by a proof in classical **ND**. Now trasform this proof into a proof in $\mathrm{ND}_{\mathcal{ALC}}$ by replacing each $\rightarrow$ by $\sqsubseteq$.

Since $\mathrm{ND}_{\mathcal{ALC}}$ is a conservative extension of the classical propositional **ND** system that has the necessitation as a derived rule, and, proves axiom

$$\forall R.(A \sqcap B) \equiv (\forall R.A \sqcap \forall R.B)$$

we have the completeness for $\mathrm{ND}_{\mathcal{ALC}}$ by a relative completeness to the axiomatic presentation of $\mathcal{ALC}$. □

Theorem 4. $\mathrm{ND}_{\mathcal{ALC}}$ *is sound regarding the standard semantics of* $\mathcal{ALC}$.

$$\textit{if} \quad \Omega \vdash \gamma \quad \textit{then} \quad \Omega \models \gamma$$

Proof. It follows direct from Lemma 5. □

Lemma 5. *Let* Π *be a deduction in* $\mathrm{ND}_{\mathcal{ALC}}$ *of* F *with all hypothesis in* $\Omega = (\mathcal{C}, \mathcal{S})$*, then if* F *is a concept:*

$$\mathcal{S} \models \left(\bigsqcap\nolimits_{A \in \mathcal{C}} A\right) \sqsubseteq F$$

and if F *is a subsumption* $A_1 \sqsubseteq A_2$*:*

$$\mathcal{S} \models \left(\bigsqcap\nolimits_{A \in \mathcal{C}} A\right) \sqcap A_1 \sqsubseteq A_2$$

For the sake of clear presentation in the following proof we adopt some special notations. The labelled concept ${}^{L}\alpha$ will be taken as equivalent to its $\mathcal{ALC}$ correspondent concept $\sigma({}^{L}\alpha)$. Letters γ and δ stand for labelled concepts while α and β stand for $\mathcal{ALC}$ concepts. We take $\mathcal{C}$ as the intersection of the concept descriptions in $\mathcal{C}$, $\bigsqcap_{A \in \mathcal{C}} A$.

Proof. The proof of Lemma 5 is done by induction on the height of the proof tree Π, represented by $\mid \Pi \mid$.

Base case If $\mid \Pi \mid = 1$ then $\Omega \vdash {}^{L}\alpha$ is such that ${}^{L}\alpha$ is in Ω. In that case, is easy to see that Lemma 5 holds since by basic set theory $(A \cap B) \subseteq A$ for all A and B.

Rule $\sqcap$-e By induction hypothesis, if $\begin{array}{c}\Pi_1\\ {}^{L}(\alpha \sqcap \beta)\end{array}$ is a derivation with all hypothesis in $\{\mathcal{C}, \mathcal{S}\}$ then $\mathcal{S} \models \mathcal{C} \sqsubseteq {}^{L}(\alpha \sqcap \beta)$. From the definition of labeled concepts and Axiom 1 we can rewrite to $\mathcal{S} \models \mathcal{C} \sqsubseteq {}^{L}\alpha \sqcap {}^{L}\beta$ which from basic set theory guarantee $\mathcal{S} \models \mathcal{C} \sqsubseteq {}^{L}\alpha$.

Rule $\sqcap$-i Let us consider the two derivations $\begin{array}{c}\Pi_1\\ {}^{L}\alpha\end{array}$ and $\begin{array}{c}\Pi_2\\ {}^{L}\beta\end{array}$ with all hypothesis in $\{\mathcal{C}_1, \mathcal{S}_1\}$ and $\{\mathcal{C}_2, \mathcal{S}_2\}$. By induction hypothesis, (1) $\mathcal{S}_1 \models \mathcal{C}_1 \sqsubseteq {}^{L}\alpha$ an (2) $\mathcal{S}_2 \models \mathcal{C}_2 \sqsubseteq {}^{L}\beta$. Now let us consider the deduction

$$\frac{\begin{array}{cc}\Pi_1 & \Pi_2\\ {}^{L}\alpha & {}^{L}\beta\end{array}}{{}^{L}(\alpha \sqcap \beta)}$$

with all hypothesis in $\{\mathcal{C}_1 \cup \mathcal{C}_2, \mathcal{S}_1 \cup \mathcal{S}_2\}$. It is easy to see that from (1) and (2) $\mathcal{S}_1 \cup \mathcal{S}_2 \models (\mathcal{C}_1 \sqcap \mathcal{C}_2) \sqsubseteq {}^{L}\alpha$ and $\mathcal{S}_1 \cup \mathcal{S}_2 \models (\mathcal{C}_1 \sqcap \mathcal{C}_2) \sqsubseteq {}^{L}\beta$. From basic set theory we may write $\mathcal{S}_1 \cup \mathcal{S}_2 \models (\mathcal{C}_1 \sqcap \mathcal{C}_2) \sqsubseteq {}^{L}\alpha \sqcap {}^{L}\beta$ and finally from Axiom 1 we get the desired result $\mathcal{S}_1 \cup \mathcal{S}_2 \models (\mathcal{C}_1 \sqcap \mathcal{C}_2) \sqsubseteq {}^{L}(\alpha \sqcap \beta)$.

Rules $\sqcup$-i Again by induction hypothesis, if $\begin{array}{c}\Pi_1\\ {}^{L}\alpha\end{array}$ is a derivation with all hypothesis in $\{\mathcal{C}, \mathcal{S}\}$ then $\mathcal{S} \models \mathcal{C} \sqsubseteq {}^{L}\alpha$. Using basic set theory we can rewrite to $\mathcal{S} \models \mathcal{C} \sqsubseteq {}^{L}\alpha \sqcup {}^{L}\beta$ and using Axiom 3 to $\mathcal{S} \models \mathcal{C} \sqsubseteq {}^{L}(\alpha \sqcup \beta)$.

Rule ($\sqcup$-e) By induction hypothesis, if

$$\begin{array}{c}\Pi_1\\ {}^{L}(\alpha \sqcup \beta),\end{array} \qquad \begin{array}{c}[{}^{L}\alpha]\\ \Pi_2\\ \gamma\end{array} \quad \text{and} \quad \begin{array}{c}[{}^{L}\beta]\\ \Pi_3\\ \gamma\end{array}$$

are derivations with hypothesis in $\{\mathcal{C}, \mathcal{S}\}$, $\{{}^{L}\alpha, \mathcal{S}\}$ and $\{{}^{L}\beta, \mathcal{S}\}$, respectively. Then, $\mathcal{S} \models \mathcal{C} \sqsubseteq {}^{L}(\alpha \sqcup \beta)$, $\mathcal{S} \models {}^{L}\alpha \sqsubseteq \gamma$ and $\mathcal{S} \models {}^{L}\beta \sqsubseteq \gamma$. From set theory $\mathcal{S} \models ({}^{L}\alpha \sqcup {}^{L}\beta) \sqsubseteq \gamma$ and from Axiom 3, $\mathcal{S} \models {}^{L}(\alpha \sqcup \beta) \sqsubseteq \gamma$. Now by the transitivity of set inclusion we can get the desired result $\mathcal{S} \models \mathcal{C} \sqsubseteq \gamma$.

Rules $\forall$-i, $\forall$-e, $\exists$-i and $\exists$-e They are sound since the premises and conclusions have the same semantics.

Rule $\neg$-e By induction hypothesis, if

$$\begin{array}{c} \Pi_1 \\ {}^{L}\alpha \end{array} \quad \text{and} \quad \begin{array}{c} \Pi_2 \\ {}^{\neg L}\neg\alpha \end{array}$$

are derivation with hypothesis in $\{\mathcal{C}_1, \mathcal{S}_1\}$ and $\{\mathcal{C}_2, \mathcal{S}_2\}$ we know that $\mathcal{S}_1 \models \mathcal{C}_1 \sqsubseteq {}^{L}\alpha$ and $\mathcal{S}_2 \models \mathcal{C}_2 \sqsubseteq {}^{\neg L}\neg\alpha$. Now consider the deduction

$$\dfrac{\begin{array}{c} \Pi_1 \\ {}^{L}\alpha \end{array} \quad \begin{array}{c} \Pi_2 \\ {}^{\neg L}\neg\alpha \end{array}}{\bot}$$

with hypothesis in $\{\mathcal{S}_1 \cup \mathcal{S}_2, \mathcal{C}_1 \cup \mathcal{C}_2\}$. By inductive hypothesis we can write $\mathcal{S}_1 \cup \mathcal{S}_2 \models \mathcal{C}_1 \sqsubseteq {}^{L}\alpha$ and $\mathcal{S}_2 \cup \mathcal{S}_2 \models \mathcal{C}_2 \sqsubseteq {}^{\neg L}\neg\alpha$. Now, from the fact that $\mathcal{ALC}$ semantics states ${}^{L}\alpha$ and ${}^{\neg L}\neg\alpha$ as two disjoint sets, we have $\mathcal{C}_1 \sqcap \mathcal{C}_2 = \emptyset$ and we can write $\mathcal{S}_1 \cup \mathcal{S}_2 \models (\mathcal{C}_1 \sqcap \mathcal{C}_2) \sqsubseteq \bot$ as desired.

Rule $\neg$-i If $\{\mathcal{C}, \mathcal{S}\}$ holds all the hypothesis of the deduction $\begin{array}{c} {}^{L}\alpha \\ \Pi_2 \\ \bot \end{array}$ then by induction hypothesis $\mathcal{S} \models \mathcal{C} \sqcap {}^{L}\alpha \sqsubseteq \bot$ (taking $\bot$ as its semantics counterpart, namely, the empty set). From basic set theory $\mathcal{S} \models \mathcal{C} \sqsubseteq {}^{\neg L}\neg\alpha$ as desired.

Rule $\bot_c$ The argument is similar from above.

Rule $\sqsubseteq$-e By induction hypothesis, if $\begin{array}{c} \Pi_1 \\ \gamma \end{array}$ and $\begin{array}{c} \Pi_2 \\ \gamma \sqsubseteq \delta \end{array}$ are deduction with hypothesis in $\{\mathcal{C}_1, \mathcal{S}_1\}$ and $\{\mathcal{C}_2, \mathcal{S}_2\}$, we have (1) $\mathcal{S}_1 \models \mathcal{C}_1 \sqsubseteq \gamma$ and (2) $\mathcal{S}_2 \models \mathcal{C}_2 \sqcap \gamma \sqsubseteq \delta$. Let us now consider the application of rule $\sqsubseteq$-e to construct the derivation

$$\dfrac{\begin{array}{c} \Pi_1 \\ \gamma \end{array} \quad \begin{array}{c} \Pi_2 \\ \gamma \sqsubseteq \delta \end{array}}{\delta}$$

with hypothesis in $\{C_1 \cup C_2, S_1 \cup S_2\}$. From (2) and $\mathcal{ALC}$ semantics we can conclude $\mathcal{S}_1 \cup \mathcal{S}_2 \models \mathcal{C}_2 \sqcap \gamma \sqsubseteq \delta$. Finally, from basic set theory $\mathcal{C}_1 \sqcap \mathcal{C}_2 \sqsubseteq \mathcal{C}_2$ we obtain $\mathcal{S}_1 \cup \mathcal{S}_2 \models \mathcal{C}_1 \sqcap \mathcal{C}_2 \sqsubseteq \delta$.

Rule $\sqsubseteq$-i By induction hypothesis, if $\begin{array}{c} \gamma \\ \Pi_1 \\ \delta \end{array}$ is a deduction with hypothesis in $\{\mathcal{C}, \mathcal{S}\}$ then $\mathcal{S} \models \mathcal{C} \sqsubseteq \delta$ and we conclude $\mathcal{S} \models \mathcal{C}^- \sqcap \gamma \sqsubseteq \delta$ where $\mathcal{C}^-$ is $\mathcal{C} - \{\gamma\}$. The semantics of $\mathcal{C}$ and $\mathcal{C}^- \sqcap \gamma$ are obviously the same.

Rule *Gen* Let Π be a proof of ${}^{L}\alpha$ following from an empty set of hypothesis, we may write $\vdash {}^{L}\alpha$. That is, ${}^{L}\alpha$ is a DL-tautology or $\sigma({}^{L}\alpha)^{\mathcal{I}} \equiv \Delta^{\mathcal{I}}$. From the necessitation rule, whenever a concept C is a DL-tautology, for any given R, the concept $\forall R.C$ will be also. For that, we can conclude that ${}^{\forall R,L}\alpha$ for any given R will be also a tautology. Remember that ${}^{\forall R,L}\alpha \equiv \forall R.\sigma({}^{L}\alpha)$. $\square$

4 Normalization theorem for $\mathrm{ND}_{\mathcal{ALC}}$

In this section we prove the normalization theorem for $\mathrm{ND}_{\mathcal{ALC}}$. It is worth nothing that the usual reductions for obtaining a normal proof in classical propositional logic also applies to $\mathrm{ND}_{\mathcal{ALC}}$. Thus, the first thing to observe is that we follow [Prawitz, 1965] approach incremented by [Seldin, 1989] permutation rules for the classical absurdity $\bot_c$. That is, using a set transformations, we move any application of $\bot_c$-rule downwards the conclusion. After this transformation we end up with a proof having in each *branch* at most one $\bot_c$-rule application as the last rule of it.

In order to move the absurdity rule downwards the conclusion and also to have a more succinct proof we restrict the language to the fragment $\{\neg, \forall, \sqcap, \sqsubseteq\}$. This will not limit our results since any $\mathcal{ALC}$ formula can be rewritten in an equivalent one in this restricted fragment. We shall consider the system $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ obtained from $\mathrm{ND}_{\mathcal{ALC}}$ by removing from $\mathrm{ND}_{\mathcal{ALC}}$ $\sqcup$-rules and $\exists$-rules. The Proposition 6 states that the system $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ is essentially just a syntactic variation of $\mathrm{ND}_{\mathcal{ALC}}$ system.

Proposition 6. *The* $\mathrm{ND}_{\mathcal{ALC}}$ $\sqcup$*-rules and* $\exists$*-rules are derived in* $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$.

Proof. Considering the concept description ${}^{L}\alpha \sqcup \beta$ being defined by ${}^{L}\neg(\neg\alpha \sqcap \neg\beta)$ and the concept description ${}^{L}\exists R.\alpha$ by ${}^{L}\neg\forall R.\neg\alpha$.

The rules ($\sqcup$-i) can be derived as follows:

$$\dfrac{\dfrac{{}^{L}\alpha \qquad \dfrac{\left[{}^{\neg L}(\neg\alpha \sqcap \neg\beta)\right]^{1}}{{}^{\neg L}\neg\alpha}\ \sqcap\text{-e}}{\bot}\ \neg\text{-e}}{{}^{L}\neg(\neg\alpha \sqcap \neg\beta)}\ \neg\text{-i} \qquad\qquad \dfrac{\dfrac{{}^{L}\beta \qquad \dfrac{\left[{}^{\neg L}(\neg\alpha \sqcap \neg\beta)\right]^{1}}{{}^{\neg L}\neg\beta}\ \sqcap\text{-e}}{\bot}\ \neg\text{-e}}{{}^{L}\neg(\neg\alpha \sqcap \neg\beta)}\ \neg\text{-i}$$

where L contains only existential quantified labels. $\neg L$ as described in Section 3, is the negation of L, that is, universal quantified are changed to existential quantified and vice versa. We note that rule $\sqcup$-i proviso requires that L contains only existential quantified labels, what makes the rule $\sqcap$-e proviso satisfied since $\neg L$ will only contains universal quantified labels. The rule $\sqcup$-e can also be derived:

$$\dfrac{\dfrac{\dfrac{\dfrac{\begin{array}{c}[{}^{L}\alpha]\\ \vdots\\ \gamma\end{array} \quad [\neg\gamma]}{\bot}}{{}^{\neg L}\neg\alpha} \qquad \dfrac{\dfrac{\begin{array}{c}[{}^{L}\beta]\\ \vdots\\ \gamma\end{array} \quad [\neg\gamma]}{\bot}}{{}^{\neg L}\neg\beta}}{{}^{\neg L}(\neg\alpha \sqcap \neg\beta)} \qquad {}^{L}\neg(\neg\alpha \sqcap \neg\beta)}{\dfrac{\bot}{\gamma}}$$

For rules ($\exists$-i) and ($\exists$-e), it is worth noting that $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ does not restrict the occurrence of existential labels, only the existential constructor of $\mathcal{ALC}$. In other words, we have just reused the $\mathcal{ALC}$ constructors $\forall$ and $\exists$ to "type" the labels and keep track of the original role quantification when it is promoted to label. Nevertheless, although the confusion could be avoided if we adopted $\neg\forall R$ instead of $\exists R$ in the labels of $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ concepts, for clear presentation we choose to allow $\exists R$ on $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ concept's labels.

$$\dfrac{\dfrac{{}^{L,\exists R}\alpha \qquad \dfrac{[{}^{\neg L}\forall R.\neg\alpha]}{{}^{(\neg L),\forall R}\neg\alpha}}{\bot}}{{}^{L}\neg\forall R.\neg\alpha} \qquad\qquad \dfrac{\dfrac{\dfrac{[{}^{(\neg L),\forall R}\neg\alpha]}{{}^{\neg L}\forall R.\neg\alpha} \qquad {}^{L}\neg\forall R.\neg\alpha}{\bot}}{{}^{L,\exists R}\alpha}$$

In the sequel, we adopt [Prawitz, 1965] terminologies such as: formula-tree, deductions or derivations, rule application, minor and major premises, *threads*, *branches* and so on. Nevertheless some terminologies have different definition in our system, in that case, we will present that definition.

A *branch* in a $\mathrm{ND}_{\mathcal{ALC}}$ or $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ deduction is an initial part $\alpha_1, \alpha_2, \ldots, \alpha_n$ of a thread such that α_n is either (i) the first formula occurrence in the thread that is a minor premise of an application of $\sqsubseteq$-e or (ii) the last formula occurrence of a thread (the end-formula of the deduction) if there is no such premise in the thread.

Given a deduction Π on $\mathrm{ND}_{\mathcal{ALC}}$ or $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$, we define the *height* of a formula occurrence α in Π inductively:

- if α is the end-formula of Π (conclusion), then $h(\alpha) = 0$;
- if α is a premise of a rule application, say λ, in Π and is not the end-formula of Π, then $h(\alpha) = h(\beta) + 1$ where β is the conclusion of λ.

In a similar matter we can define the height of a *rule application* in a deduction.

A *maximal formula* is a formula occurrence that is consequence of an introduction rule and the major premise of an elimination rule. A maximal $\sqsubseteq$-formula in a proof Π is a maximal formula that is a subsumption.

Lemma 7. *Let Π be a proof of α (concept or subsumption of concepts) from Δ in $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$. Then there is a proof Π' without maximal $\sqsubseteq$-formulas.*

Proof. We prove Lemma 7 by induction over the number of maximal $\sqsubseteq$-formulas occurrences. We apply a sequence of reductions choosing always a highest maximal $\sqsubseteq$-formula occurrence in the proof tree. In the reduction shown below we note that α cannot be a subsumption, so that, the reduction application will never introduce new maximal $\sqsubseteq$-formulas. In other words, we cannot have nested subsumptions, subsumptions are not concepts.

$$\dfrac{\begin{array}{c}\Pi_1\\ \alpha\end{array} \quad \dfrac{\begin{array}{c}[\alpha]\\ \Pi_2\\ \beta\end{array}}{\alpha \sqsubseteq \beta}}{\beta} \qquad \rhd \qquad \begin{array}{c}\Pi_1\\ [\alpha]\\ \Pi_2\\ \beta\end{array}$$

Lemma 8 Moving $\perp_c$ downwards on branches. *If $\Omega \vdash_{\mathrm{ND}^{-}{}_{\mathcal{ALC}}} \alpha$, then there is a deduction Π in $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ of α from Ω where each branch in Π has at most one application of $\perp_c$-rule and, whenever it has one, it is one of the following cases: (i) the last rule applied in this branch; (ii) its conclusion is the premise of a $\sqsubseteq$-i application, being this $\sqsubseteq$-i the last rule applied in the branch.*

Proof. Let Π be a deduction in $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ of α (subsumption of concepts or concept) from a set of hypothesis Δ. Let λ be an application of a $\perp_c$-rule in Π with $h(\lambda) = d$ such that there is no other application of $\perp_c$-rule above λ. Let us consider each possible rule application immediately below λ. For each case, we show how one can exchange the rules decreasing the height of λ.

Rule $\forall$-e

$$\dfrac{\dfrac{\begin{array}{c}[{}^{\neg L}\neg\forall R.\alpha]\\ \vdots\\ \perp\end{array}}{{}^{L}\forall R.\alpha}}{{}^{L,\forall R}\alpha} \qquad \rhd \qquad \dfrac{\begin{array}{c}\dfrac{\dfrac{\dfrac{[{}^{L}\forall R.\alpha]}{{}^{L,\forall R}\alpha} \quad [{}^{\neg L,\exists R}\neg\alpha]}{\perp}}{{}^{\neg L}\neg\forall R.\alpha}\\ \vdots\\ \perp\end{array}}{{}^{L,\forall R}\alpha}$$

Rule $\forall$-i

$$
\dfrac{\dfrac{\begin{matrix}[{}^{\neg L,\exists R}\neg\alpha]\\ \vdots \\ \bot\end{matrix}}{{}^{L,\forall R}\alpha}}{{}^{L}\forall R.\alpha}
\quad \rhd \quad
\dfrac{\begin{matrix}\dfrac{\dfrac{[{}^{L,\forall R}\alpha]}{{}^{L}\forall R.\alpha} \quad [{}^{\neg L}\neg\forall R.\alpha]}{\dfrac{\bot}{{}^{\neg L,\exists R}\neg\alpha}}\\ \vdots \\ \bot\end{matrix}}{{}^{L}\forall R.\alpha}
$$

Rule $\sqcap$-i

$$
\dfrac{\dfrac{\begin{matrix}{}^{\exists L}\neg\alpha\\ \vdots \\ \bot\end{matrix}}{{}^{\forall L}\alpha} \quad \begin{matrix}\Pi\\ {}^{\forall L}\beta\end{matrix}}{{}^{\forall L}(\alpha\sqcap\beta)}
\quad \rhd \quad
\dfrac{\begin{matrix}\dfrac{\dfrac{[{}^{\forall L}\alpha]^{2} \quad \begin{matrix}\Pi\\ {}^{\forall L}\beta\end{matrix}}{{}^{\forall L}(\alpha\sqcap\beta)} \quad [{}^{\exists L}\neg(\alpha\sqcap\beta)]^{1}}{\dfrac{\bot}{{}^{\exists L}\neg\alpha}\,2}\\ \vdots \\ \bot\end{matrix}}{{}^{\forall L}(\alpha\sqcap\beta)}\,1
$$

Rule $\sqcap$-e

$$
\dfrac{\dfrac{\begin{matrix}{}^{\exists L}\neg(\alpha\sqcap\beta)\\ \vdots \\ \bot\end{matrix}}{{}^{\forall L}(\alpha\sqcap\beta)}}{{}^{\forall L}\alpha}
\quad \rhd \quad
\dfrac{\begin{matrix}\dfrac{[{}^{\exists L}\neg\alpha]^{2} \quad \dfrac{[{}^{\forall L}(\alpha\sqcap\beta)]^{1}}{{}^{\forall L}\alpha}}{\dfrac{\bot}{{}^{\exists L}\neg(\alpha\sqcap\beta)}\,1}\\ \vdots \\ \bot\end{matrix}}{{}^{\forall L}\alpha}\,2
$$

Rule $\neg$-e

$$
\dfrac{\dfrac{\begin{matrix}[{}^{\neg L}\neg\alpha]\\ \vdots \\ \bot\end{matrix}}{{}^{L}\alpha} \quad \begin{matrix}[\Delta]\\ \Pi\\ {}^{\neg L}\neg\alpha\end{matrix}}{\bot}
\quad \rhd \quad
\begin{matrix}\dfrac{[{}^{L}\alpha] \quad \begin{matrix}[\Delta]\\ \Pi\\ {}^{\neg L}\neg\alpha\end{matrix}}{\dfrac{\bot}{{}^{\neg L}\neg\alpha}}\\ \vdots \\ \bot\end{matrix}
$$

One must observe that in all reductions above, the conclusion of $\bot_c$ rule application is the premise of the rule considered in each case. That is why the $\neg$-i rule was not considered, if so, the conclusion of $\bot_c$ rule would have to be a $\bot$, wish is prohibit by the restriction on $\bot_c$-rule.

Rule $\sqsubseteq$-e

$$\dfrac{\begin{array}{c}[\neg\alpha]\\ \Pi_1\\ \dfrac{\bot}{\alpha}\end{array} \quad \begin{array}{c}\Pi_2\\ \alpha\sqsubseteq\beta\end{array}}{\beta} \quad \rhd \quad \begin{array}{c}\dfrac{\dfrac{[\alpha]^1 \quad \begin{array}{c}\Pi_2\\ \alpha\sqsubseteq\beta\end{array}}{\beta} \quad [\neg\beta]^2}{\dfrac{\bot}{\neg\alpha}\,1}\\ \Pi_1\\ \dfrac{\bot}{\beta}\,2\end{array}$$

The reductions below will be used in the induction step in Theorem 9.

Let Π be a deduction of α from Ω which contains a maximal formula occurrence F. We say that Π' is a reduction of Π at F if we obtain Π' by removing F using the reductions below. Since F clearly can not be atomic, each reduction refers to a possible principal sign of F. If the principal sign of F is ψ, then Π' is said to be a ψ-reduction of Π. In each case, one can easily verify that Π' obtained is still a deduction of α from Ω.

$\sqcap$-reduction

$$\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ {}^{\forall L}\alpha\end{array} \quad \begin{array}{c}\Pi_2\\ {}^{\forall L}\beta\end{array}}{{}^{\forall L}(\alpha\sqcap\beta)}}{{}^{\forall L}\alpha} \quad \rhd \quad \begin{array}{c}\Pi_1\\ {}^{\forall L}\alpha\end{array}$$

$\forall$-reduction

$$\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ {}^{L,\forall R}\alpha\end{array}}{{}^{L}\forall R.\alpha}}{{}^{L,\forall R}\alpha} \quad \rhd \quad \begin{array}{c}\Pi_1\\ {}^{L,\forall R}\alpha\end{array}$$

$\neg$-reduction

$$\dfrac{\dfrac{\begin{array}{c}[{}^{L}\alpha]\\ \Pi_1\\ \bot\end{array}}{{}^{\neg L}\neg\alpha} \quad \begin{array}{c}\Pi_2\\ {}^{L}\alpha\end{array}}{\bot} \quad \rhd \quad \begin{array}{c}\Pi_2\\ [{}^{L}\alpha]\\ \Pi_1\\ \bot\end{array}$$

The fact that DL has no concept internalizing $\sqsubseteq$ imposes quite particular features on the form of the normal proofs in $\mathrm{ND}_{\mathcal{ALC}}$.

A $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ deduction is called *normal* when it does not have maximal formula occurrences. Theorem 9 shows how we can construct a normal deduction in $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$.

Consider a deduction Π in $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$. Applying Lemma 7 we obtain a new deduction Π' without any maximal $\sqsubseteq$-formulas. Then we apply Lemma 8 to reduce the number of applications of $\perp_c$-rule on each branch and moving the remaining downwards to the end of each branch. Without loss of generality we can from now on consider any deduction in $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ as having no maximal $\sqsubseteq$-formula and at most one $\perp_c$-rule application per branch, namely, the last one application in the branch.

Theorem 9 normalization of $\mathrm{ND}_{\mathcal{ALC}}$**.** *If $\Omega \vdash_{\mathrm{ND}^{-}{}_{\mathcal{ALC}}} \alpha$, then there is a normal deduction in $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ of α from Ω.*

Proof. Let Π be a deduction in $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$ having the form remarked in the previous paragraph. Consider the pair (d, n) where d is the maximum degree among the maximal formulas, and n is the number of maximal formulas with degree d. We proceed the normalization proof by induction on the lexicographic pair (d, n).

Let F be one of the highest maximal formula with degree d and consider each possible case according the principal sign of F.

If F has as principal sign $\sqcap$, applying the $\sqcap$-reduction we get a new deduction Π_1 with complexity (d_1, n_1). We now have $d_1 \leq d$, depending on the existence of other maximal $\sqcap$-formulas on Π. If $d_1 = d$, then necessarily $n_1 < n$. The cases where the principal sign of F is $\neg$ or $\forall$ are similar. Two facts can be observed. First, the $\sqsubseteq$-reduction will not be used anymore, since Π does not have any remaining maximal $\sqsubseteq$-formula. Second, although the $\neg$-reduction can increase the number of maximal formulas, those maximal formulas will undoubtedly have degree less than d, so that, we indeed have $(d_1, n_1) < (d, n)$. So, by the induction hypothesis, we have that Π_1 is normalizable and so is Π for each principal sign considered.

As we have already mentioned $\mathrm{ND}_{\mathcal{ALC}}$ has no concept internalization $\sqsubseteq$. This imposes quite particular form of the normal proofs in $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$. Consider a thread in a deduction Π in $\mathrm{ND}^{-}{}_{\mathcal{ALC}}$, such that no element of the thread is a minor premise of $\sqsubseteq$-e rule. We shall see that if Π is normal, the thread can be divided into two parts. There is one formula occurrence A in the thread such that all formula occurrences in the thread above A are premises of applications of elimination rules and all formula occurrences below A in the thread (except the last one) are premises of applications of introduction rules. Therefore, in the first part of the thread, we start from the top-most formula an decrease the complexity of that until A. In the second part of the thread we pass to more and more complex formulas. Given that, A is said thus the minimum formula in the thread. Moreover, each branch on Π has at most one application of $\perp_c$ rule as its last rule application.

Normalization is important since from it one can provide complete procedure to produce canonical proofs in $\mathcal{ALC}$. Canonical proofs are important regarding explaining theorem hood.

5 Dealing with $\mathcal{ALCQI}$ theories

One of the main goals of this article is to show how $\mathrm{ND}_{\mathcal{ALCQI}}$ facilitates the reasoning explanation on UML reasoning. To illustrate this in real cases, we will need to move to a more expressive DL. In fact, from [Berardi et al., 2005], [Calvanese et al., 1998b], [Calvanese et al., 1998a], [Calvanese et al., 2009], and [Calvanese et al., 2004] we know that in order to express UML modeling and reasoning, we have to use $\mathcal{ALCQI}$. It is $\mathcal{ALC}$ with number restrictions and inverse roles. That is, we extend the language adding the following constructors:

$$\begin{array}{l} C ::= \bot \mid A \mid \neg C \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \exists R.C \mid \forall R.C \mid \leq nR.C \mid \geq nR.C \\ R ::= P \mid P^- \end{array}$$

where A stands for atomic concepts and P for atomic roles. Some of the above operators can be mutually defined: (i) $\bot$ for $A \sqcap \neg A$; (ii) $\top$ for $\neg\bot$; (iii) $\geq kR.C$ for $\neg(\leq k-1R.C)$; (iv) $\leq kR.C$ for $\neg(\geq k+1R.C)$; (v) $\exists R.C$ for $\geq 1R.C$.

An $\mathcal{ALCQI}$ theory is a finite set of inclusion assertions of the form $C_1 \sqsubseteq C_2$. The semantics of $\mathcal{ALCQI}$ constructors and theories is analogous to that of $\mathcal{ALC}$. The semantics of the new constructors, inverse roles and qualified number restrictions, are as follows:

$$\begin{array}{r} (P^-)^{\mathcal{I}} = \{(a, a') \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid (a', a) \in P^{\mathcal{I}}\} \\ (\leq kR.C)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \#\{a' \in \Delta^{\mathcal{I}} \mid (a, a') \in R^{\mathcal{I}} \wedge a' \in C^{\mathcal{I}}\} \leq k\} \end{array}$$

The **ND** for $\mathcal{ALCQI}$, named $\mathrm{ND}_{\mathcal{ALCQI}}$, is presented in Figure 5. The system is a conservative extension of $\mathrm{ND}_{\mathcal{ALC}}$ and each new introduced rule is sound, so it is also sound. $\mathrm{ND}_{\mathcal{ALCQI}}$ normalization is work in progress. For the main purpose of this article, completeness does not matter. Anyway, a completeness proof for $\mathrm{ND}_{\mathcal{ALCQI}}$ follows from a (technically heavy) mapping from a complete Sequent Calculus for $\mathcal{ALCQI}$ to $\mathrm{ND}_{\mathcal{ALCQI}}$. The notation used in Figure 5 is similar of that used in Section 3. In some rules, we superscribe the list of labels with the kind of labels allowed on it. For example, in rule $\sqcap$-e, we restrict L to contain only $\forall$ and $\geq n$ labels using the notation $L^{\forall\geq}$. Moreover, for easier understanding, some provisos regarding the order relation between the number n and m are presented on the right of the rule's name.

A normalization proof for $\mathrm{ND}_{\mathcal{ALCQI}}$ is obtained as an extension of the normalization proof presented for $\mathcal{ALC}$ in Section 4. For instance, reductions as the

$$\frac{{}^{L^{\forall\geq}}(\alpha \sqcap \beta)}{{}^{L^{\forall\geq}}\alpha}\ \sqcap\text{-e} \qquad \frac{{}^{L^{\forall\geq}}(\alpha \sqcap \beta)}{{}^{L^{\forall\geq}}\beta}\ \sqcap\text{-e} \qquad \frac{{}^{L^{\forall\leq}}\alpha \quad {}^{L^{\forall\leq}}\beta}{{}^{L^{\forall\leq}}(\alpha \sqcap \beta)}\ \sqcap\text{-i}$$

$$\frac{{}^{L^{\exists\leq}}(\alpha \sqcup \beta) \qquad \begin{matrix}[{}^{L^{\exists\leq}}\alpha] \\ \vdots \\ \gamma\end{matrix} \qquad \begin{matrix}[{}^{L^{\exists\leq}}\beta] \\ \vdots \\ \gamma\end{matrix}}{\gamma}\ \sqcup\text{-e} \qquad \frac{{}^{L^{\exists\geq}}\alpha}{{}^{L^{\exists\geq}}(\alpha \sqcup \beta)}\ \sqcup\text{-i} \qquad \frac{{}^{L^{\exists\geq}}\beta}{{}^{L^{\exists\geq}}(\alpha \sqcup \beta)}\ \sqcup\text{-i}$$

$$\frac{{}^{L^{\forall\exists}}\alpha \qquad \neg {}^{L^{\forall\exists}}\neg\alpha}{\perp}\ \neg\text{-e} \qquad \frac{\begin{matrix}[{}^{L^{\forall\exists}}\alpha] \\ \vdots \\ \perp\end{matrix}}{\neg {}^{L^{\forall\exists}}\neg\alpha}\ \neg\text{-i} \qquad \frac{\begin{matrix}[\neg {}^{L^{\forall\exists}}\neg\alpha] \\ \vdots \\ \perp\end{matrix}}{{}^{L^{\forall\exists}}\alpha}\ \perp_c$$

$$\frac{{}^{L}\exists R.\alpha}{{}^{L,\exists R}\alpha}\ \exists\text{-e} \qquad \frac{{}^{L,\exists R}\alpha}{{}^{L}\exists R.\alpha}\ \exists\text{-i} \qquad \frac{{}^{L}\forall R.\alpha}{{}^{L,\forall R}\alpha}\ \forall\text{-e}$$

$$\frac{{}^{L,\forall R}\alpha}{{}^{L}\forall R.\alpha}\ \forall\text{-i} \qquad \frac{{}^{L}\leq nR.\alpha}{{}^{L,\leq nR}\alpha}\ \leq\text{-e} \qquad \frac{{}^{L,\leq nR}\alpha}{{}^{L}\leq nR.\alpha}\ \leq\text{-i}$$

$$\frac{{}^{L}\geq nR.\alpha}{{}^{L,\geq nR}\alpha}\ \geq\text{-e} \qquad \frac{{}^{L,\geq nR}\alpha}{{}^{L}\geq nR.\alpha}\ \geq\text{-i}$$

$$\frac{{}^{\exists R,L}\alpha}{{}^{\geq 1R,L}\alpha}\ \geq\exists \qquad \frac{{}^{\geq nR,L}\alpha}{{}^{\exists R,L}\alpha}\ \exists\geq (n \geq 1)$$

$$\frac{{}^{\geq mR,L}\alpha}{{}^{\geq nR,L}\alpha}\ \text{dec } (m \geq n) \qquad \frac{{}^{\leq mR,L}\alpha}{{}^{\leq nR,L}\alpha}\ \text{inc } (m \leq n) \qquad \frac{{}^{L}\alpha}{{}^{\forall R,L}\alpha}\ Gen$$

$$\frac{{}^{L_1}\alpha \qquad {}^{L_1}\alpha \sqsubseteq {}^{L_2}\beta}{{}^{L_2}\beta}\ \sqsubseteq\text{-e} \qquad \frac{\begin{matrix}[{}^{L_1}\alpha] \\ \vdots \\ {}^{L_2}\beta\end{matrix}}{{}^{L_1}\alpha \sqsubseteq {}^{L_2}\beta}\ \sqsubseteq\text{-i} \qquad \frac{{}^{\exists R,L_1}\alpha \sqsubseteq {}^{L_2}\beta}{{}^{L_1}\alpha \sqsubseteq {}^{\forall R^-,L_2}\beta}\ inv$$

Figure 5: The Natural Deduction system for $\mathcal{ALCQI}$

following

$$\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ {}^{L_1,\leq nR}\alpha^{L_2}\end{array}}{{}^{L_1}\!\leq nR.\alpha^{L_2}}}{{}^{L_1,\leq nR}\alpha^{L_2}} \quad \rhd \quad \begin{array}{c}\Pi_1\\ {}^{L_1,\leq nR}\alpha^{L_2}\end{array}$$

have to be considered. The presentation of the complete normalization proof for $\mathcal{ALCQI}$ is out of the scope of this paper since it can be considered as an article in its own.

Normalization provides a proof procedure for $\mathrm{ND}_{\mathcal{ALCQI}}$. Initially decompose the (candidate) conclusion ($\alpha \sqsubseteq \beta$) by means of introduction rules applied bottom-up, until atomic labeled concepts. For each atomic concept, one chooses an hypothesis from Δ and by decomposing it, by means of elimination rules, tries to achieve this very atomic (labeled) concept. This allows us to derive a (complete) proof procedure for the logic, decomposing the conclusions and the hypothesis until atomic levels an proving one set using the other. In our case we are interested in applying this proof procedure on top of theories.

Moreover, since theories must be closed under generalizations, we introduce the following Generalization rules in order to reflect this closure.

$$\frac{\alpha \sqsubseteq \beta}{{}^{\forall R}\alpha \sqsubseteq {}^{\forall R}\beta}\,\mathrm{G}\forall \qquad \frac{\alpha \sqsubseteq \beta}{{}^{\exists R}\alpha \sqsubseteq {}^{\exists R}\beta}\,\mathrm{G}\exists$$

$$\frac{\alpha \sqsubseteq \beta}{{}^{\leq nR}\alpha \sqsubseteq {}^{\leq nR}\beta}\,\mathrm{G}\!\leq \qquad \frac{\alpha \sqsubseteq \beta}{{}^{\geq nR}\alpha \sqsubseteq {}^{\geq nR}\beta}\,\mathrm{G}\!\geq$$

6 Explaining UML in $\mathrm{ND}_{\mathcal{ALCQI}}$

In [Berardi et al., 2005], DLs are used to formalize UML diagrams. It uses two DL languages: $\mathcal{DLR}_{ifd}$ or $\mathcal{ALCQI}$. The diagram on Figure 6 and its formalization on Figure 7, are from [Berardi et al., 2005].

We use examples of DL deductions from [Berardi et al., 2005, page 84], using $\mathrm{ND}_{\mathcal{ALCQI}}$ to reason on the $\mathcal{ALCQI}$ KB. The idea is to exemplify how one can obtain from $\mathrm{ND}_{\mathcal{ALCQI}}$ proofs, a more precise and direct explanation.

The first example concerns a refinement of a multiplicity. That is, from reasoning on the diagram, one can deduce that the class `MobileCall` participates on the association `MobileOrigin` with multiplicity $0\ldots1$, instead of the $0\ldots*$ presented in the diagram. The proof on $\mathrm{ND}_{\mathcal{ALCQI}}$ is as follows, where we abbreviate the class names for their first letters, for instance, `Origin` (`O`), `MobileCall` (`MC`), `call` (`c`) and so on. Note that $\neg \geq 2\mathtt{c}^{-}.\mathtt{MO}$ is actually an abbreviation for $\leq 1\mathtt{c}^{-}.\mathtt{MO}$.

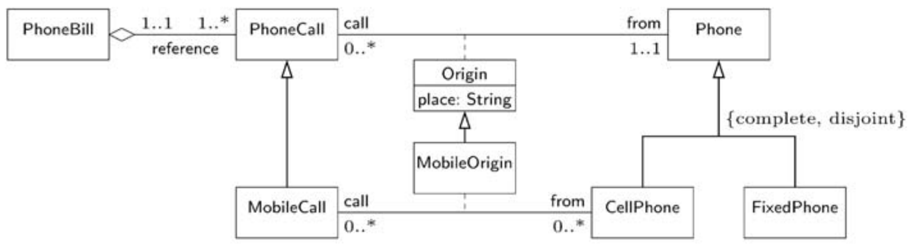

Figure 6: UML class diagram

$$
\dfrac{\dfrac{[\geq 2\ \mathtt{c}^-.\mathtt{MO}]^2 \quad \dfrac{\mathtt{MO} \sqsubseteq \mathtt{O}}{\geq 2\ \mathtt{c}^-.\mathtt{MO} \sqsubseteq\ \geq 2\ \mathtt{c}^-.\mathtt{O}}\,\mathrm{G}\geq}{\geq 2\ \mathtt{c}^-.\mathtt{O}}\,\sqsubseteq\text{-e} \qquad \dfrac{\dfrac{\dfrac{[\mathtt{MC}]^1 \quad \mathtt{MC} \sqsubseteq \mathtt{PC}}{\mathtt{PC}}\,\sqsubseteq\text{-e} \quad \mathtt{PC} \sqsubseteq\ \geq 1\ \mathtt{c}^-.\mathtt{O} \sqcap \leq 1\ \mathtt{c}^-.\mathtt{O}}{\geq 1\ \mathtt{c}^-.\mathtt{O} \sqcap \leq 1\ \mathtt{c}^-.\mathtt{O}}\,\sqsubseteq\text{-e}}{\leq 1\ \mathtt{c}^-.\mathtt{O}}\,\sqcap\text{-e}}{\dfrac{2\ \dfrac{\bot}{\neg \geq 2\ \mathtt{c}^-.\mathtt{MO}}\,\neg\text{-i}}{1\ \mathtt{MC} \sqsubseteq \neg \geq 2\ \mathtt{c}^-.\mathtt{MO}}\,\sqsubseteq\text{-i}}\,\neg\text{-e}
$$

To exemplify deductions on diagrams, an incorrect generalization between two classes was introduced. The generalization asserts that each `CellPhone` is a `FixedPhone`, which means the introduction of the new axiom `CellPhone` $\sqsubseteq$ `FixedPhone` in the KB. From that improper generalization, several undesirable properties could be drawn.

The first conclusion about the modified diagram is that `Cellphone` is now inconsistent. The $\mathrm{ND}_{\mathcal{ALCQI}}$ proof below explicits that from the newly introduced axiom and from the axiom `CellPhone` $\sqsubseteq \neg$`FixedPhone` in the KB, one can conclude that `CellPhone` is now inconsistent.

$$
1\ \dfrac{\dfrac{\dfrac{\mathtt{Cell} \sqsubseteq \neg\mathtt{Fixed} \quad [\mathtt{Cell}]^1}{\neg\mathtt{Fixed}}\,\sqsubseteq\text{-e} \quad \dfrac{\mathtt{Cell} \sqsubseteq \mathtt{Fixed} \quad [\mathtt{Cell}]^1}{\mathtt{Fixed}}\,\sqsubseteq\text{-e}}{\bot}\,\neg\text{-e}}{\mathtt{Cell} \sqsubseteq \bot}\,\sqsubseteq\text{-i}
$$

The second conclusion is that in the modified diagram, `Phone` $\equiv$ `FixedPhone`. Note that we have only to show that `Phone` $\sqsubseteq$ `FixedPhone` since `FixedPhone` $\sqsubseteq$ `Phone` is an axiom already in the original KB. We can conclude from the proof below that `Phone` $\sqsubseteq$ `FixedPhone` is not a direct consequence of `CellPhone` being inconsistent, as stated in [Berardi et al., 2005], but it is mainly a direct conse-

$$
\begin{aligned}
\texttt{Origin} &\sqsubseteq \forall\texttt{place.String}\\
\texttt{Origin} &\sqsubseteq \exists\texttt{place.}\top \sqcap (\leq 1\ \texttt{place})\\
\texttt{Origin} &\sqsubseteq \exists\texttt{call.PhoneCall} \sqcap (\leq 1\ \texttt{call}) \sqcap \exists\texttt{from.Phone} \sqcap (\leq 1\ \texttt{from})\\
\texttt{MobileOrigin} &\sqsubseteq \exists\texttt{call.MobileCall} \sqcap (\leq 1\texttt{call}) \sqcap \exists\texttt{from.CellPhone} \sqcap (\leq 1\ \texttt{from})\\
\texttt{PhoneCall} &\sqsubseteq (\geq 1\ \texttt{call}^{-}\texttt{.Origin}) \sqcap (\leq 1\ \texttt{call}^{-}\texttt{.Origin})\\
\top &\sqsubseteq \forall\texttt{reference}^{-}\texttt{.PhoneBill} \sqcap \forall\texttt{reference.PhoneCall}\\
\texttt{PhoneBill} &\sqsubseteq (\geq 1\ \texttt{reference}^{-})\\
\texttt{PhoneCall} &\sqsubseteq (\geq 1\ \texttt{reference}) \sqcap (\leq 1\ \texttt{reference})\\
\texttt{MobileCall} &\sqsubseteq \texttt{PhoneCall}\\
\texttt{MobileOrigin} &\sqsubseteq \texttt{Origin}\\
\texttt{CellPhone} &\sqsubseteq \texttt{Phone}\\
\texttt{FixedPhone} &\sqsubseteq \texttt{Phone}\\
\texttt{CellPhone} &\sqsubseteq \neg\texttt{FixedPhone}\\
\texttt{Phone} &\sqsubseteq \texttt{CellPhone} \sqcup \texttt{FixedPhone}
\end{aligned}
$$

Figure 7: The $\mathcal{ALCQI}$ theory obtained from the UML diagram on Figure 6

quence of the newly introduced axiom and a cases analysis over the possible subtypes of `Phone`.

$$
1\ \dfrac{\dfrac{\dfrac{[\texttt{Phone}]^{1} \qquad \texttt{Phone} \sqsubseteq \texttt{Cell} \sqcup \texttt{Fixed}}{\texttt{Cell} \sqcup \texttt{Fixed}}\sqsubseteq\text{-e} \qquad \dfrac{[\texttt{Cell}] \qquad \texttt{Cell} \sqsubseteq \texttt{Fixed}}{\texttt{Fixed}}\sqsubseteq\text{-e} \qquad [\texttt{Fixed}]}{\texttt{Fixed}}\sqcup\text{-e}}{\texttt{Phone} \sqsubseteq \texttt{Fixed}}\sqsubseteq\text{-i}
$$

Below it is shown the above discussed subsumption proved in **SC**, based on the system presented in [Rademaker & Haeusler, 2008].

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\texttt{MO} \Rightarrow \texttt{O}}{\geq 2\ call^{-}.\texttt{MO} \Rightarrow\ \geq 2\ call^{-}.\texttt{O}}}{\texttt{MC}, \geq 2\ call^{-}.\texttt{MO} \Rightarrow\ \geq 2\ call^{-}.\texttt{O}} \qquad \dfrac{\dfrac{\texttt{MC} \Rightarrow \texttt{PC} \qquad \texttt{PC} \Rightarrow\ \geq 1\ call^{-}.\texttt{O} \sqcap \leq 1\ call^{-}.\texttt{O}}{\texttt{MC} \Rightarrow\ \geq 1\ call^{-}.\texttt{O} \sqcap \leq 1\ call^{-}.\texttt{O}}}{\texttt{MC}, \geq 2\ call^{-}.\texttt{MO} \Rightarrow\ \geq 1\ call^{-}.\texttt{O} \sqcap \leq 1 call^{-}.\texttt{O}}}{\texttt{MC}, \geq 2\ call^{-}.\texttt{MO} \Rightarrow\ \geq 1\ call^{-}.\texttt{O} \sqcap \leq 1 call^{-}.\texttt{O} \sqcap \geq 2 call^{-}.\texttt{O}}}{\texttt{MC}, \geq 2\ call^{-}.\texttt{MO} \Rightarrow \bot}}{\texttt{MC} \Rightarrow \neg \geq 2\ call^{-}.\texttt{MO}}
$$

In order to the reader concretely see that it is harder explaining on Tableaux basis than on Natural Deduction basis, we prove the same $\texttt{MC} \sqsubseteq \neg \geq 2\ \texttt{call}^{-}.\texttt{MO}$ subsumption in Tableaux. We follow [Baader et al., 2003, Section 2.3.2.1] and represent the Tableaux constraints as **ABox** assertions without unique name assumption. [8] The constraint"a belongs to (the interpretation of) C" is represented by $C(a)$ and "b is an R-filler of a" by $R(a,b)$. The interested reader can find the complete presentation of the Tableaux procedure for $\mathcal{ALCQI}$ in [Baader et al., 2003].

The Tableaux procedure starts translating the subsumption problem to a satisfiability problem. The subsumption $C \sqsubseteq D$ holds if and only if $C \sqcap \neg D$ is unsatisfiable. In our case, $C_0 \equiv \texttt{MC} \sqcap \geq 2\ \texttt{call}^{-}.\texttt{MO}$ should be unsatisfiable. Since C_0 is already in the NNF (negation normal form), we are ready to the Tableaux algorithm, otherwise we would have to first transform it to obtain a NNF equivalent concept description. Tableaux procedure starts with the ABox $A_0 = \{C_0(x_0)\}$ and applies consistency-preserving transformation rules to the ABox until no more rules apply. If the completed expanded ABox obtained does not contain clashes (contradictory assertions), then A_0 is consistent and thus C_0 is satisfiable, and inconsistent (unsatisfiable) otherwise.

$\mathcal{A}_0$ is the initial ABox. By $\sqcap$-rule, we get $\mathcal{A}_1$. Than, by $\geq$-rule we get $\mathcal{A}_2$. $\mathcal{A}_3$ is obtained by using the theory axioms $\texttt{MO} \sqsubseteq \texttt{O}$ and $\texttt{MC} \sqsubseteq \texttt{PC}$. The ABox $\mathcal{A}_4$ is obtained by using the theory axiom $\texttt{PC} \sqsubseteq \geq 1\ \mathit{call}^{-}.\texttt{O} \sqcap \leq 1\ \mathit{call}^{-}.\texttt{O}$. Next, $\mathcal{A}_5$ by $\sqcap$-rule. ABox $\mathcal{A}_5$ now contains a contradiction, the individual a is required to have at most one successor of type $\texttt{O}$ in the role call^{-}. Nevertheless, b and c are also required to be of type $\texttt{O}$ and successors of a in role call^{-}, vide $\mathcal{A}_3$ and $\mathcal{A}_2$. This shows that C_0 is unsatisfiable, and thus $\texttt{MC} \sqsubseteq \neg \geq 2\ \texttt{call}^{-}.\texttt{MO}$.

$$\{(\texttt{MC} \sqcap \geq 2\ \texttt{call}^{-}.\texttt{MO})(a)\} \qquad (\mathcal{A}_0)$$

$$\mathcal{A}_0 \cup \{\texttt{MC}(a), (\geq 2\ \mathit{call}^{-}.\texttt{MO})(a)\} \qquad (\mathcal{A}_1)$$

$$\mathcal{A}_1 \cup \{\mathit{call}^{-}(a,b), \mathit{call}^{-}(a,c), \texttt{MO}(b), \texttt{MO}(c), a \neq b, b \neq c, a \neq c\} \qquad (\mathcal{A}_2)$$

$$\mathcal{A}_2 \cup \{\texttt{O}(b), \texttt{O}(c), \texttt{PC}(a)\} \qquad (\mathcal{A}_3)$$

$$\mathcal{A}_3 \cup \{(\geq 1\ \mathit{call}^{-}.\texttt{O} \sqcap \leq 1\ \mathit{call}^{-}.\texttt{O})(a)\} \qquad (\mathcal{A}_4)$$

$$\mathcal{A}_4 \cup \{(\geq 1\ \mathit{call}^{-}.\texttt{O})(a), (\leq 1\ \mathit{call}^{-}.\texttt{O})(a)\} \qquad (\mathcal{A}_5)$$

7 Conclusion

We presented ND systems for $\mathcal{ALC}$ and $\mathcal{ALCQI}$ and showed, by means of some examples, how it can be useful to explain formal facts on theories obtained from UML models. Instead of UML, ER could also be used according a similar framework. Regarding the examples used in this article and the explanations obtained,

[8] Instead, we allow explicit inequality assertions of the form $x \neq y$. Those assertions are assumed symmetric.

it is worthwhile noting that the Natural Deduction proofs obtained are quite close to the natural language explanation provided by the authors of the article from which the examples are taken. It is an easy task to provide the respective natural language explanation for a comparison. This article shows that **ND** deduction systems are better than Tableaux and Sequent Calculus as structures to be used in explaining theorem when validating theories in the presence of false positives. We also remark and show how normalization is important in order to provide well-structured proofs.

References

[Baader et al., 2003] Baader, F., Calvanese, D., McGuinness, D., Nardi, D., & Patel-Schneider, P., Eds. (2003). *The Description Logic Handbook*. Cambridge University Press, first edition.

[Berardi et al., 2005] Berardi, D., Calvanese, D., & De Giacomo, G. (2005). Reasoning on UML class diagrams. *Artificial Intelligence*, 168(1-2), 70–118.

[Borgida et al., 1999] Borgida, A., Franconi, E., Horrocks, I., McGuinness, D., & Patel-Schneider, P. (1999). Explaining $\mathcal{ALC}$ subsumption. In *Proceddings of the International Workshop on Description Logics* (pp. 33–36).

[Calvanese et al., 2009] Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., & Rosati, R. (2009). Conceptual modeling for data integration. In A. Borgida, V. Chaudhri, P. Giorgini, & E. Yu (Eds.), *John Mylopoulos Festschrift*, volume 5600 of *Lecture Notes in Computer Science*. Springer. To appear.

[Calvanese et al., 1998a] Calvanese, D., De Giacomo, G., Lenzerini, M., Nardi, D., & Rosati, R. (1998a). Information integration: Conceptual modeling and reasoning support. In *Proc. of the 6th International Conference on Cooperative Information Systems (CoopIS'98)* (pp. 280–291).

[Calvanese et al., 2004] Calvanese, D., Giacomo, G. D., Lenzerini, M., Rosati, R., & Vetere, G. (2004). DL-Lite: Practical reasoning for rich DLs. In *Proc. of the 2004 Description Logic Workshop, DL 2004*, volume 104 of *CEUR Electronic Workshop Proceedings, http://ceur-ws.org*.

[Calvanese et al., 1998b] Calvanese, D., Lenzerini, M., & Nardi, D. (1998b). Description logics for conceptual data modeling. In J. Chomicki & G. Saake (Eds.), *Logics for Databases and Information Systems* (pp. 229–263). Kluwer Academic Publisher.

[Gentzen, 1935] Gentzen, G. (1935). Untersuchungen über das logische schließen. *Mathematische Zeitschrift*, 39, 176–210.

[Haeusler & Rademaker, 2009] Haeusler, E. H. & Rademaker, A. (2009). Is it important to explain a theorem? a case study on uml and $\mathcal{ALCQI}$. In *ER '09: Proceedings of the ER 2009 Workshops (CoMoL, ETheCoM, FP-UML, MOST-ONISW, QoIS, RIGiM, SeCoGIS) on Advances in Conceptual Modeling - Challenging Perspectives* (pp. 34–44). Berlin, Heidelberg: Springer-Verlag.

[Liebig & Halfmann, 2005] Liebig, T. & Halfmann, M. (2005). Explaining subsumption in $\mathcal{ALEHF}_{R^+}$ tboxes. In I. Horrocks, U. Sattler, & F. Wolter (Eds.), *Proc. of the 2005 International Workshop on Description Logics, DL 2005* (pp. 144–151). Edinburgh, Scotland.

[Maibaum, 2005] Maibaum, T. S. E. (2005). The epistemology of validation and verification testing. In *Testing of Communicating Systems, 17th IFIP TC6/WG 6.1 International Conference, TestCom 2005* (pp. 1–8). Montreal, Canada.

[McGuinness, 1996] McGuinness, D. (1996). *Explaining Reasoning in Description Logics*. PhD thesis, Rutgers University.

[Prawitz, 1965] Prawitz, D. (1965). *Natural deduction: a proof-theoretical study*. PhD thesis, Almqvist & Wiksell.

[Rademaker & Haeusler, 2008] Rademaker, A. & Haeusler, E. H. (2008). Toward short and structural $\mathcal{ALC}$-reasoning explanations: A sequent calculus approach. In *Proceedings of Brazilian Symposium on Artificial Inteligence. Advances in Artificial Intelligence – SBIA 2008* (pp. 167–176).: Springer-Verlag. `http://dx.doi.org/10.1007/978-3-540-88190-2_22`.

[Schild, 1991] Schild, K. (1991). A correspondence theory for terminological logics: preliminary report. In *IJCAI'91: Proceedings of the 12th international joint conference on Artificial intelligence* (pp. 466–471). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. Also published at TR 91, Technische Universitat Berlin.

[Seldin, 1989] Seldin, J. (1989). Normalization and excluded middle. I. *Studia Logica*, 48(2), 193–217.

[Smullyan, 1968] Smullyan, R. (1968). *First-Order Logic*. Springer-Verlag.

[Takeuti, 1975] Takeuti, G. (1975). *Proof Theory. Number 81 in Studies in Logic and the Foundations of Mathematics*. North-Holland.