A Labelled Natural Deduction System for an Intuitionistic Description Logic with Nominals

Bernardo **Alkmim**[1], Edward **Haeusler**[1] and Cláudia **Nalon**[2]

[1]*Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Brazil*
[2]*University of Brasília (UnB), Brazil*

Abstract

In this paper we present a new labelled Natural Deduction calculus for the logic iALC, an intuitionistic description logic with nominals originally designed to reason over laws and other normative sentences in general. Even though this logic already has a formalised Sequent Calculus system, Natural Deduction is more adequate for extracting explanations for non-logicians - which is further aided by our use of labels as part of the calculus. We prove soundness and normalisation for the new Natural Deduction system, and show its completeness.

Keywords

Natural Deduction, Description Logic, Intuitionistic Logic, iALC

1. Introduction

iALC is an intuitionistic description logic with nominals originally designed to model and reason over norms - especially laws and other forms of juridical norms. It is based on the framework for intuitionistic modal logic **IK** [1, 2, 3]. These modal logics arise from interpreting the usual possible worlds definitions in an intuitionistic metatheory. The logic language also borrows some of its design principles from [4] where the framework IHL, for intuitionistic hybrid logics, is introduced. The syntax and semantics of iALC are given in Section 2.

iALC is a combination of logics, inheriting its syntax from description logics and its semantics from intuitionistic logic. The description logic component allows for modelling normative sentences whilst avoiding many of the problems that occur in non-modal frameworks when considering affirmative and normative sentences as the same thing. The intuitionistic aspect is based on Kelsenian Jurisprudence, which revolves around the concept of *ground norms*. Norms can be stratified in such a way that ground norms precede other individual laws (known as legal individuals and usually referred as Valid Legal Statements, or VLSs), thus creating a hierarchy of coexisting different legal systems. In iALC, a VLS is represented by a world in a Kripke model - and a nominal in the language.

As shown in [5], the intuitionistic aspect allows for non-conflicting existence of thought-to-be contradictory logical formulas that can be modelled in a more natural way than than

---

in Description Logics (DLs) that have a classical notion of validity, such as ALC and other canonical DLs. In the domain of law, this allows for the existence of a model representing two distinct legal systems, for instance, one in which the death penalty is not allowed and the other in which it is (referencing the same concept of death penalty). Representing law in a classical DL would force us to find a more conflated way to insert legal individuals into the logic at the risk of losing legibility and, even worse, soundness. In the example of death penalty, one would need a concept $DeathPenaltyBrazil$ and another $DeathPenaltyTexas$ as well as one or more formulas connecting both concepts in one manner or another (and would have to find another way to model VLSs, instead of directly having them being worlds in the model). By having this intuitionistic perception, we better translate the foundations of Private International Law into the modelling, connecting VLSs which are part of different legal systems via the same concept, for example, $DeathPenalty$, without having to resort to many workarounds. Legal precedence is, as well, something defined constructively in the logic through the VLSs, and does not need additional formulations in the model. In a regular (i.e. classical) DL, this cannot be represented as directly as in its intuitionistic counterpart when modelling laws since a Kripke model for a Classical Logic collapses all worlds (the legal individuals) to the same one, causing a contradiction by allowing and prohibiting the death penalty at the same time.

The logic iALC has been compared to other logical languages with respect to its adequacy for representing normative systems. Comparison between iALC and different flavours of Deontic Logics [6, 7, 8, 9] have been discussed in [10, 5, 11, 12]. There are also other works, such as [13, 14], in which the logics are bound to the *closed world assumption*, which may not be adequate for dealing with legal systems, where unknown facts cannot be considered untrue. In [15], the presented logic utilises classical negation, despite being very similar to iALC. With classical negation, different legal systems would be in the complement of others automatically, not allowing them to have some subset of laws that is the same (at least when considering a direct modelling approach as the one allowed in iALC).

A Sequent Calculus (SC) for iALC was previously presented [10, 16, 17]. However, later works argued that ND systems lead to better readability and explainability for non-logicians when compared to the SC [11, 12, 18] and the Hilbert system to be introduced in Section 3. ND is an answer provided by Jaskowiski in 1929 to a challenge offered by Lukasiewicz in his 1926 seminars. He observed that mathematicians do not use axiomatic systems to prove theorems, using, instead, assumptions and higher patterns of deduction, normally stated in natural language. As an epistemological argument in favour of the naturality of ND, not only Jaskowski, but also Gentzen, independently proposed quite similar systems of ND. Of course, Gentzen's motivation was his proof-theoretical approach to the consistency proof, which, nevertheless was only possible, according to his own words, to the definition of SC, a kind of meta-calculus for ND. We believe that this very motivation for reflecting the logical form of reasoning that mathematicians usually employ is enough motivation to consider ND as a good tool to represent formal arguments to people.

In this paper, we formally introduce the Natural Deduction system for iALC for the first time, bridging the gap between its informal presentation and the new calculus for iALC together with correctness results. It is our hope that the results contained in this paper will help fostering the utilisation of iALC not only as a language for the representation of the intended problems related to laws, but also to the implementation of tools for reasoning about those problems

whilst retaining human-readability.

In the calculus, we introduce the use of labels as syntactic artefacts in order to represent contexts to formulas of iALC. We assign labels in a similar fashion to [19], where the author gives a labelled ND system for the classical sibling of iALC, $\mathcal{ALC}$. Labels represent *contexts* for formulas in iALC, and allow for better concept manipulation. The use of labels was made essential in [11, 12, 5], where, even in an informal introduction to a ND system for iALC, we noticed it needed a way to carry context over from formula to formula, especially when formulas involved multiple universal or existential restrictions.

The structure of the paper is as follows. In Section 2, we introduce the logic itself. In Section 3 we present the new Natural Deduction system for iALC, explaining the design choices we made. In Section 4 we show the normalisation, completeness and soundness proofs for the system. Finally, in Section 5, we conclude and discuss future work.

2. The Logic iALC

The logic iALC, first introduced in [20] and further developed in [16, 17, 10], is an intuitionistic description logic with nominals created to represent and reason about legal knowledge. It was used to answer multiple-choice questions from the Brazilian Bar Exam in [11], to formalise Brazilian Law in [12], and to model and deal with problems involving trust, privacy and transparency in knowledge graphs in [5], showing its adequacy to deal with legal representation and reasoning. We formally introduce its language in the following:

Let α and β be concepts, A be an atomic concept, R be an atomic role, δ be any formula and x be a nominal. iALC formulas are described by the following grammar:

$$\delta ::= \alpha \mid x : \alpha$$

where the concepts α, β are given by the following grammar:

$$\alpha, \beta ::= A \mid \bot \mid \top \mid \neg\alpha \mid \alpha \sqcap \beta \mid \alpha \sqcup \beta \mid \alpha \sqsubseteq \beta \mid \exists R.\alpha \mid \forall R.\alpha$$

where A is an atomic concept. Formulas have a restricted use of nominals as nominal assertions are not concept constructors.

As given by the definition above, the main difference from the grammar of ALC is that, in iALC, $\sqsubseteq$ is a concept-forming operator, whereas in $\mathcal{ALC}$ it is not, as defined and explained in [16]: Because iALC has an intuitionistic semantics, this entailment operator is analogous to the intuitionistic implication. This construction allows for an abstraction of the precedence relation $\preceq$ (a pre-order on worlds, as in the Kripke semantics for intuitionistic logics) in TBox reasoning, as well as allowing us to define negation using $\sqsubseteq$.

A constructive interpretation of iALC is a structure $\mathcal{I}$ consisting of a non-empty set $\Delta^{\mathcal{I}}$ of entities in which each entity represents a partially defined individual; a refinement pre-ordering $\preceq^{\mathcal{I}}$ on $\Delta^{\mathcal{I}}$, i.e. a reflexive and transitive relation (a pre-order), and an interpretation function $\cdot^{\mathcal{I}}$ mapping each role name R to a binary relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ and atomic concept A to a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ which is closed under refinement, i.e. $x^{\mathcal{I}} \in A^{\mathcal{I}}$ and $x^{\mathcal{I}} \preceq^{\mathcal{I}} y^{\mathcal{I}}$ implies $y^{\mathcal{I}} \in A^{\mathcal{I}}$.

The logic follows the semantics of **IK**, where the structures $\mathcal{I}$ are models for iALC if they satisfy two frame conditions:

F1 if $x \preceq x'$ and xRy then $\exists y'.x'Ry'$ and $y \preceq y'$; and

F2 if $y \preceq y'$ and xRy then $\exists x'.x'Ry'$ and $x \preceq x'$.

The interpretation $\mathcal{I}$ is lifted from atomic concepts to arbitrary concepts as follows:

$$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$$

$$\bot^{\mathcal{I}} = \varnothing$$

$$(\alpha \sqcap \beta)^{\mathcal{I}} = \alpha^{\mathcal{I}} \cap \beta^{\mathcal{I}}$$

$$(\alpha \sqcup \beta)^{\mathcal{I}} = \alpha^{\mathcal{I}} \cup \beta^{\mathcal{I}}$$

$$(\neg\alpha)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \forall y, x \preceq y \Rightarrow y \notin \alpha^{\mathcal{I}}\}$$

$$(\forall R.\alpha)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \forall y(x \preceq y \Rightarrow \forall z((y, z) \in R^{\mathcal{I}} \Rightarrow z \in \alpha^{\mathcal{I}}))\}$$

$$(\exists R.\alpha)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \forall y(x \preceq y \Rightarrow \exists z((y, z) \in R^{\mathcal{I}} \wedge z \in \alpha^{\mathcal{I}}))\}$$

$$(\alpha \sqsubseteq \beta)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \forall y, (x \preceq y \wedge y \in \alpha^{\mathcal{I}}) \Rightarrow y \in \beta^{\mathcal{I}}\}$$

And the interpretation for formulas with nominals such as $(x : \delta)^{\mathcal{I}}$ (where δ is any formula) is $x^{\mathcal{I}} \models_{\mathcal{I}} \delta^{\mathcal{I}}$, where $\models_{\mathcal{I}}$ is the usual satisfaction relation of Kripke models.

3. Natural Deduction

The ND system we developed aims to reach TBox validity and it uses Gentzen-style proofs, based especially on [21]. Formulas are labelled in order to allow for the mixing of nominals and existential/universal restrictions in a TBox-only style of reasoning, even though role assertions belong in the ABox.

We expand the language with labels for our ND system, essentially giving context of universal and existential restrictions in formulas, as explained in Section 1. The grammar for iALC is then extended with:

$$\alpha ::= \alpha^L$$

$$L ::= \exists R, L \mid \forall R, L \mid \varnothing$$

As for labelled formulas, following [19], we first define an function σ to make a labelled ND system that translates from a non-labelled formula to a labelled formula as given below:

$$\sigma(\alpha) = \alpha^{\varnothing}$$

$$\sigma(\forall R.\alpha) = \sigma(\alpha)^{\forall R}$$

$$\sigma(\exists R.\alpha) = \sigma(\alpha)^{\exists R}$$

$$\sigma(\alpha \sqcap \beta) = \sigma(\alpha) \sqcap \sigma(\beta)$$

$$\sigma(\alpha \sqcup \beta) = \sigma(\alpha) \sqcup \sigma(\beta)$$

$$\sigma(\neg\alpha) = \neg\, \sigma(\alpha)$$

$$\sigma(\alpha \sqsubseteq \beta) = \sigma(\alpha) \sqsubseteq \sigma(\beta)$$

Given σ, the semantics for a labelled formula δ is given by $\sigma(\delta)^{\mathcal{I}}$. This way, $\alpha^{\forall R_3, \exists R_2, \forall R_1}$, for example, has the same semantics as $\forall R_1.\exists R_2.\forall R_3.\alpha$.

3.1. The Rules

Table 1
Rules of the ND System

$$
\begin{array}{c}
[\alpha^{L1}] \\
\vdots \\
\dfrac{\beta^{L2}}{\alpha^{L1} \sqsubseteq \beta^{L2}} \; \sqsubseteq -i
\end{array}
\qquad
\dfrac{\alpha^{L1} \quad \alpha^{L1} \sqsubseteq \beta^{L2}}{\beta^{L2}} \; \sqsubseteq -e
\qquad
\begin{array}{c}
[\alpha^{L}] \\
\vdots \\
\dfrac{\bot}{(\neg\alpha)^{\neg L}} \; \neg -i
\end{array}
$$

$$
\dfrac{\alpha^{L} \quad (\neg\alpha)^{\neg L}}{\bot} \; \bot -i
\qquad
\dfrac{\alpha^{L^\forall} \quad \beta^{L^\forall}}{(\alpha \sqcap \beta)^{L^\forall}} \; \sqcap -i
\qquad
\dfrac{(\alpha \sqcap \beta)^{L^\forall}}{\alpha^{L^\forall}} \; \sqcap -e_1
\qquad
\dfrac{(\alpha \sqcap \beta)^{L^\forall}}{\beta^{L^\forall}} \; \sqcap -e_2
$$

$$
\dfrac{\alpha^{L^\exists}}{(\alpha \sqcup \beta)^{L^\exists}} \; \sqcup -i_1
\qquad
\dfrac{\beta^{L^\exists}}{(\alpha \sqcup \beta)^{L^\exists}} \; \sqcup -i_2
\qquad
\begin{array}{c}
\qquad\quad [\alpha^{L^\exists}] \quad [\beta^{L^\exists}] \\
\qquad\quad \vdots \qquad\;\; \vdots \\
\dfrac{(\alpha \sqcup \beta)^{L^\exists} \quad \delta \qquad \delta}{\delta} \; \sqcup -e
\end{array}
$$

$$
\dfrac{y : \alpha^{\exists R,L}}{x : (\exists R.\alpha)^L} \; \exists -i
\qquad
\dfrac{x : (\exists R.\alpha)^L}{y : \alpha^{\exists R,L}} \; \exists -e
\qquad
\dfrac{y : \alpha^{\forall R,L}}{x : (\forall R.\alpha)^L} \; \forall -i
\qquad
\dfrac{x : (\forall R.\alpha)^L}{y : \alpha^{\forall R,L}} \; \forall -e
$$

$$
\dfrac{\bot}{\delta} \; efq
\qquad
\begin{array}{c}
[x : \alpha^{L1}] \\
\vdots \\
\dfrac{x : \beta^{L2}}{x : (\alpha^{L1} \sqsubseteq \beta^{L2})} \; \sqsubseteq -i_n
\end{array}
\qquad
\dfrac{x : \alpha^{L1} \quad x : (\alpha^{L1} \sqsubseteq \beta^{L2})}{x : \beta^{L2}} \; \sqsubseteq -e_n
$$

$$
\begin{array}{c}
[x : \alpha^{L}] \\
\vdots \\
\dfrac{x : \bot}{x : (\neg\alpha)^{\neg L}} \; \neg -i_n
\end{array}
\qquad
\dfrac{x : \alpha^{L} \quad x : (\neg\alpha)^{\neg L}}{x : \bot} \; \bot -i_n
\qquad
\dfrac{x : \alpha^{L^\forall} \quad x : \beta^{L^\forall}}{x : (\alpha \sqcap \beta)^{L^\forall}} \; \sqcap -i_n
$$

$$
\dfrac{x : (\alpha \sqcap \beta)^{L^\forall}}{x : \alpha^{L^\forall}} \; \sqcap -e_{1_n}
\qquad
\dfrac{x : (\alpha \sqcap \beta)^{L^\forall}}{x : \beta^{L^\forall}} \; \sqcap -e_{2_n}
\qquad
\dfrac{x : \alpha^{L^\exists}}{x : (\alpha \sqcup \beta)^{L^\exists}} \; \sqcup -i_{1_n}
$$

$$
\dfrac{x : \beta^{L^\exists}}{x : (\alpha \sqcup \beta)^{L^\exists}} \; \sqcup -i_{2_n}
\qquad
\begin{array}{c}
\qquad\quad [x : \alpha^{L^\exists}] \quad [x : \beta^{L^\exists}] \\
\qquad\quad \vdots \qquad\qquad \vdots \\
\dfrac{x : (\alpha \sqcup \beta)^{L^\exists} \quad \delta \qquad\qquad \delta}{\delta} \; \sqcup -e_n
\end{array}
\qquad
\dfrac{x : \bot}{\delta} \; efq_n
$$

$$
\dfrac{\alpha^{L}}{\alpha^{L,\forall R}} \; Gen
\qquad
\dfrac{x : \alpha^{L}}{y : \alpha^{L,\forall R}} \; Gen_n
\qquad
\dfrac{x : (\exists R.\alpha)^{L1} \quad x : \forall R.(\alpha^{L1} \sqsubseteq \beta^{L2})}{x : (\exists R.\beta)^{L2}} \; p - \exists
$$

Table 1 contains all rules of the calculus. Rules are presented in the usual way, where premises for each rule are given over the bar and its (single) conclusion lies below. Assumptions in the premises are enclosed by square brackets. The proofs in ND have a tree-like form, by connecting together the deduction steps given by the applied rules in the proof. Some of the rules have formulas surrounded by brackets, indicating that the rule in question can *discharge* these

hypotheses, i.e. the conclusion of the proof does not depend on them explicitly anymore.

Let α and β be concepts, x and y be nominals, δ be a formula, and R be a role. L represents a (possibly empty) list of labels. Labels represent universal or existential restrictions on concepts, made *implicit* by the introduction rules - the associated elimination rules make them *explicit*. We denote by $L^\forall$ a list that has only universal restrictions, and by $L^\exists$ a list with existential restrictions only. $\neg L$ swaps existential and universal restrictions throughout L.

Rules $\exists - i$, $\exists - e$, $\forall - i$ and $\forall - e$ require that there be an assertion xRy in order to function. These rules are what allows our calculus to use the labels as restrictions. Even though this is a TBox-centered calculus, we assume that there is a subjacent ABox extension that introduces these role assertions. In [5] there are running examples.

Rules efq and efq_n stand for the principle of explosion. The structural rules Gen and Gen_n deal with modal necessitation. Rule $p - \exists$ allows for context permutation within a subsumption.

Some rules have two versions, one without nominals and one with nominals (notated with a subscript n in the name). For instance, we have $\sqcap - i$ and $\sqcap - i_n$.

From now on, every rule that has a version with nominals (i.e. has a subscript n) will have only this version being used, as the other version is trivial in the correctness proofs given in the next section. As such, they will only be called by their original name, without the n, in order to avoid visual clutter. For instance, $\sqcap - i_n$ will be referred to as $\sqcap - i$.

4. Main Properties

In this section we show that important properties which are expected of ND systems hold in the one here proposed for iALC. Due to limited size and redundancy in arguments, some proofs are sketched and inductive cases may be omitted - mainly the ones involving operators analogue to propositional ones, since those are very similar in functionality. Full proofs can be found in https://github.com/Bpalkmim/iALC_ND_proofs. In order to improve readability, lists of labels are omitted whenever possible throughout Section 4.1, as they do not interfere with the process of normalisation. When dealing with soundness and completeness in Sections 4.2 and 4.3, however, labels will be explicitly shown.

Our main focus here is to show that these properties still hold when dealing with labels and the rules which introduce and eliminate existential and universal restrictions.

4.1. Normalisation

Normalisation is a crucial step to show that the sub-formula principle holds in our system. This principle states thatevery step in a proof contains only sub-formulas of either the conclusion or the premises. This is extremely important to order to do proof search in an efficient way. We will base our normalisation proof on the ones presented in [21, 22, 23, 24]. We will utilise the words proof and derivation interchangeably from now on (one can see a proof in ND as a derivation from the premises to the conclusion).

To reach normalisation, the following definitions are needed.

Definition 1 (*Major premise*). *The major premise is the premise of an elimination rule which contains the operator to be eliminated. Other premises, if existent, are called minor premises.*

Definition 2 (*Detour*). *A detour occurs when a formula is introduced via an introduction rule only to be eliminated immediately later as the major premise of an elimination rule.*

Our goal in the end will be to remove such detours via *reductions* and arrive at so called *normal* proofs, to show that the system has normalisation.

Example 1 (A detour with $\sqcap$). *Let Π_1 and Π_2 be sub-proofs. An example of a detour is given below, where the list of labels is empty:*

$$\dfrac{\dfrac{\begin{matrix}\Pi_1 & \Pi_2 \\ x : \alpha & x : \beta\end{matrix}}{x : (\alpha \sqcap \beta)} \;\sqcap - i}{x : \alpha} \;\sqcap - e_1$$

By introducing and eliminating a conjunction, $x : \alpha$ occurs twice in the same branch of the proof. This is clearly an unnecessary step, as no progress is made in the proof.

Maximum formulas are precisely those that represent a detour. Its definition is given next.

Definition 3 (Maximum Formula). *A formula δ in a derivation Π is a maximum formula if:*

1. *δ is the conclusion of an application of an introduction rule and the major premise of an application of an elimination rule of the same operator; or*
2. *δ is the conclusion of an application of $\sqcup - e$ as well as the major premise of an application of an elimination rule.*

The second case covers a *hidden* detours in the $\sqcup - e$. An example will be shown later.

In order to evaluate how a reduction works in simplifying a proof, it is necessary to have some kind of measure. We then introduce the concept of degree of a formula, indicating its size by the number of occurrences of logical operators. Not surprisingly, the name *maximum formula* is related to this measurement.

Definition 4 (Degree of a Formula). *The degree of a formula is given as follows:*

$$deg(C) = 0, \text{ for an atomic concept } C$$
$$deg(x : \alpha) = deg(\alpha) + 2$$
$$deg(\alpha \sqcup \beta) = deg(\alpha) + deg(\beta) + 1$$
$$deg(\alpha \sqcap \beta) = deg(\alpha) + deg(\beta) + 1$$
$$deg(\alpha \sqsubseteq \beta) = deg(\alpha) + deg(\beta) + 1$$
$$deg(\neg\alpha) = deg(\alpha) + 1$$
$$deg(\exists R.\alpha) = deg(\alpha) + 2$$
$$deg(\forall R.\alpha) = deg(\alpha) + 2$$

Definition 5 (Degree of a Derivation). *The degree of a derivation Π is defined as:*

$$deg(\Pi) = max\{deg(\delta) : \delta \text{ is a maximum formula in } \Pi\}$$

Definition 6 (Normal Derivation). *A derivation Π is called normal when $deg(\Pi) = 0$.*

In other words, a normal derivation has a maximum formula of degree 0, i.e. has no detours.

Definition 7 (Critical Derivation). *A derivation Π is called critical when*

1. *Π ends with an application of an elimination rule ρ;*
2. *the major premise δ of ρ is a maximum formula;*
3. *$deg(\Pi) = deg(\delta)$; and*
4. *for all Π' sub-derivation of Π, $deg(\Pi') < deg(\delta) = deg(\Pi)$.*

This way we know that the main detour to be reduced lies (at first) in ρ.

A reduction is an operation on proofs that functions as a way to remove maximum formulas locally. The reductions for the *propositionally-based* operators are extensions of the usual Prawitz-reductions. As such, we have a permutation operation in order to arrange a $\sqcup$-maximum formula to be properly reduced. This will be shown after the more basic cases.

Definition 8 (Reduction). *We say a derivation Π reduces to another, Π', when we apply zero or more reductions to maximum formulas of Π and it becomes equal to Π'. We denote by $\Pi \triangleright \Pi'$ the reduction from Π to Π'.*

As a quick note on reductions, they usually end up copying whole sections of proofs (thus increasing the size of the proof itself), but they do not increase the complexity of the formulas therein. First we will show some basic cases.

$\sqcap$-**reduction** Taking the derivation from Example 1, we have:

$$
\begin{array}{cc}
\begin{array}{c}
\dfrac{\dfrac{\begin{array}{cc} \Pi_1 & \Pi_2 \\ x:\alpha & x:\beta \end{array}}{x:(\alpha \sqcap \beta)} \sqcap - i}{x:\alpha} \sqcap - e_1 \\
\Pi_3
\end{array}
&
\quad \triangleright \quad
\begin{array}{c}
\Pi_1 \\
x:\alpha \\
\Pi_3
\end{array}
\end{array}
$$

This way the proof goes directly through $x : \alpha$, without introducing and eliminating an unused operator. The reduction utilises $\sqcap - e_1$, but $\sqcap - e_2$ is analogous.

$\sqsubseteq$-**reduction** This derivation has a maximum formula whose main operator is $\sqsubseteq$:

$$
\begin{array}{cc}
\begin{array}{c}
\dfrac{\begin{array}{cc} \Pi_1 \\ x:\alpha \end{array} \quad \dfrac{\begin{array}{c} [x:\alpha] \\ \Pi_2 \\ x:\beta \end{array}}{x:(\alpha \sqsubseteq \beta)} \sqsubseteq -i}{x:\beta} \sqsubseteq -e \\
\Pi_3
\end{array}
&
\quad \triangleright \quad
\begin{array}{c}
\Pi_1 \\
x:\alpha \\
\Pi_2 \\
x:\beta \\
\Pi_3
\end{array}
\end{array}
$$

Note that, in this case, there can be multiple occurrences of the hypothesis discharge over Π_2, so this reduction ends up copying the entirety of Π_1 possibly many times, increasing the size of the proof itself, but not its complexity, since the maximum formula was removed.

The case of $\neg$-reduction via $\neg - i$ and $\bot - i$ (which acts as $\neg - e$) is analogous, as negation is but a special case of $\sqsubseteq$.

$\forall$-**reduction** The case for the universal restriction on concepts is rather simple. Let R be a role. As always, we assume that xRy is in the ABox.

$$
\begin{array}{ccc}
\dfrac{\dfrac{\dfrac{\Pi_1}{y : \alpha^{\forall R}}}{x : \forall R.\alpha}\;\forall - i}{\dfrac{y : \alpha^{\forall R}}{\Pi_2}}\;\forall - e & & \dfrac{\dfrac{\Pi_1}{y : \alpha^{\forall R}}}{\Pi_2}
\end{array}
$$
$$\rhd$$

The case for $\exists$-reduction is the analogous.

$\sqcup$-**reduction** This operator, much like $\sqcap$, has two possible reductions, but both are equivalent. Only the one involving $\sqcup - i_1$ will be shown.

$$
\begin{array}{cc}
\dfrac{\dfrac{\dfrac{\Pi_1}{x : \alpha}}{x : \alpha \sqcup \beta}\;\sqcup - i_1 \quad \dfrac{[x : \alpha]}{\begin{array}{c}\Pi_2\\\delta\end{array}} \quad \dfrac{[x : \beta]}{\begin{array}{c}\Pi_3\\\delta\end{array}}}{\dfrac{\delta}{\Pi_4}}\;\sqcup - e & \dfrac{\dfrac{\dfrac{\Pi_1}{x : \alpha}}{\begin{array}{c}\Pi_2\\\delta\end{array}}}{\Pi_4}
\end{array}
$$
$$\rhd$$

As was the case with $\sqsubseteq$, there can be multiple copies of Π_1 to be added.

$\sqcup$-**permutation** This operator has a peculiarity in that it may *hide* a maximum formula, since δ can be the major premise of another elimination rule. Let us, then, define a $\sqcup$-permutation that allows for a rearrangement of the proof tree around the $\sqcup - e$. Here we will only show the case of $\sqcup - i_1$, without loss of generality. Let ρ be an elimination rule with major premise δ_1, minor premise δ_2 and conclusion δ_3. Due to limited space, we assume that ρ has only one minor premise; with more premises the argument is analogous.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\Pi_1}{x : \alpha}}{x : \alpha \sqcup \beta}\;\sqcup - i_1 \quad \dfrac{[x : \alpha][x : \beta]}{\begin{array}{c}\Pi_2\quad\Pi_3\\\delta_1\quad\delta_1\end{array}}}{\delta_1}\;\sqcup - e \quad \dfrac{\Pi_4}{\delta_2}}{\dfrac{\delta_3}{\Pi_5}}\;\rho
$$
$$\rhd$$
$$
\dfrac{\dfrac{\dfrac{\Pi_1}{x : \alpha}}{x : \alpha \sqcup \beta}\;\sqcup - i_1 \quad \dfrac{\dfrac{[x : \alpha]}{\begin{array}{c}\Pi_2\;\Pi_4\\\delta_1\;\delta_2\end{array}}}{\delta_3}\;\rho \quad \dfrac{\dfrac{[x : \beta]}{\begin{array}{c}\Pi_3\;\Pi_4\\\delta_1\;\delta_2\end{array}}}{\delta_3}\;\rho}{\dfrac{\delta_3}{\Pi_5}}\;\sqcup - e
$$

With this derivation in hand, the detour that may happen with ρ becomes apparent, since ρ is an elimination rule and there may be other detours to be reduced in the sub-derivations. It is important to note that this permutation does not increase the degree of the derivation, since the maximum formula, be it $x : \alpha \sqcup \beta$ or any of the δ's, does not increase in complexity.

In order to optimise space, from here on we will refer to simple derivations utilising the following notation: Π/δ represents a derivation Π with δ as a conclusion. Conversely, δ/Π represents a derivation Π with δ as a premise. Note that there may be more copies of δ in the actual derivation Π, as stated previously.

Lemma 1. *Let* Π_1/δ *and* δ/Π_2 *be two derivations in the ND system such that* $deg(\Pi_1) = n_1$ *and* $deg(\Pi_2) = n_2$. *Then,* $deg(\Pi_1/[\delta]/\Pi_2) = max\{deg(\delta), n_1, n_2\}$.

Proof. Directly from Definition 5: the maximum formula in this derivation is either in Π_1, in Π_2 or is δ itself. $\square$

Lemma 2. *If Π reduces to Π', then $deg(\Pi') \leq deg(\Pi)$.*

Proof. It follows directly from the reductions in Definition 8 and the degree of a formula in Definition 4: Since a reduction always lowers the degree of a maximum formula, it also lowers the degree of the derivation in which it is contained (as per Definition 5). □

Lemma 3. *Let δ be a formula in iALC, Γ be a set of formulas, $\Delta \subseteq \Gamma$ and Π be a critical derivation of $\Gamma \vdash_{ND} \delta$. Then, Π reduces to a derivation Π' of $\Delta \vdash_{ND} \delta$ such that $deg(\Pi') < deg(\Pi)$.*

Proof. The proof follows by induction on the structure of Π. We will use the reductions shown in Definition 8, but assuming they are critical derivations. We will call each derivation before a reduction Π, and Π' after the reduction. □

Lemma 4. *Let δ be a formula in iALC, Γ be a set of formulas, $\Delta \subseteq \Gamma$ and Π be a derivation of $\Gamma \vdash_{ND} \delta$ such that $deg(\Pi) > 0$. Then, Π reduces to a derivation Π' of $\Delta \vdash_{ND} \delta$ such that $deg(\Pi') < deg(\Pi)$.*

Proof. It follows directly from Lemmas 2 and 3 using induction on the length of Π. □

Theorem 1 (Normalisation). *Let Π be a derivation of $\Gamma \vdash_{ND} \delta$. Then, Π reduces to a normal derivation Π' of $\Delta \subseteq \Gamma \vdash_{ND} \delta$.*

Proof. This follows directly from Lemma 4 by applying the reductions (and the permutation when necessary) inductively over the degree of Π in order to remove its maximum formulas. In the end, the proof generated will be normal. Moreover, since it is done inductively, the normalisation process is terminating. □

4.2. Soundness

Theorem 2 (Soundness). *Let δ be a formula in iALC and Γ a set of formulas. Then, $\Gamma \vdash_{ND} \delta$ implies $\Gamma \models \delta$.*

Proof. The proof follows by showing that each rule preserves soundness. For rules that involve hypothesis discharge, we use induction on the length of derivations. □

4.3. Completeness

Theorem 3 (Completeness). *Let δ be a formula in iALC and Γ a set of formulas. Then, $\Gamma \models \delta$ implies $\Gamma \vdash_{ND} \delta$.*

Proof. We prove completeness in regards to TBox reasoning. In [17, 10], the authors provide a Hilbert system that implements TBox validity for iALC as per [1, 2, 3, 25] consisting of the following axioms and inference rules:

> (IPL) all axioms of intuitionistic propositional logic
> ($\forall$K) $(\forall R.(\alpha \sqsubseteq \beta)) \sqsubseteq (\forall R.\alpha \sqsubseteq \forall R.\beta)$
> ($\exists$K) $(\forall R.(\alpha \sqsubseteq \beta)) \sqsubseteq (\exists R.\alpha \sqsubseteq \exists R.\beta)$

$$(\text{DIST})\ \exists R.(\alpha \sqcup \beta) \sqsubseteq (\exists R.\alpha \sqcup \exists R.\beta)$$

$$(\text{DIST0})\ \exists R.\bot \sqsubseteq \bot$$

$$(\text{DISTm})\ (\exists R.\alpha \sqsubseteq \forall R.\beta) \sqsubseteq \forall R.(\alpha \sqsubseteq \beta)$$

(Nec) If α is a theorem, then $\forall R.\alpha$ is a theorem too.

(MP) If α and $\alpha \sqsubseteq \beta$ are theorems, then β is a theorem too.

In [25], the authors prove that this system is sound and complete for TBox validity. So, in order to prove completeness of our ND system, all we have to do is prove each of these axioms. Axiom (IPL) is easily proven since all substitution instances of IPL theorems can be proven in our ND system using only propositional rules. (MP) is covered by our $\sqsubseteq -e$ rule, and (Nec), by the Gen rule. We, then, only have to prove the remaining five axioms. Full proofs can be found in https://github.com/Bpalkmim/iALC_ND_proofs. $\square$

5. Final Remarks

In this paper we presented a Natural Deduction system for the intuitionistic description logic iALC, originally tailored to deal with norms, especially in a legal context. Although iALC had a Sequent Calculus (SC) system formalised, it was not widely used due to the interdisciplinary characteristic of the application of this logic, which favoured a more readable style of making proofs - Natural Deduction (ND).

The ND system is sound and complete and its normalisation is terminating. The proofs for those properties closely follow the ones presented previously for the SC system, with roots on the ND system designed for classical $\mathcal{ALC}$, so they share many similarities.

With the formalisation of this system, we intend to further apply iALC to contexts involving law and norms, and expand its usage to different areas, now with a more approachable style of showing proofs for non-logicians.

A version of this system without nominals would be PSPACE-complete. We believe that he labelled ND system for iALC with nominals has 2-EXPTIME-hard complexity. We intend to formally establish this using the same techniques given in [17, 26], which were used for the SC system for iALC. Establishing complexity is a future work.

We intend, in the future, to implement these proofs in an interactive theorem prover such as Coq and plan to create an automated theorem prover for iALC. In the theoretical side, we aim to increment iALC with some kind of non-monotonic reasoning, which will most likely be related to default logic.

Acknowledgments

References

[1] A. K. Simpson, The Proof Theory and Semantics of Intuitionistic Modal Logic, Ph.D. thesis, University of Edinburgh, UK, 1994. URL: http://hdl.handle.net/1842/407.

[2] G. Fischer-Servi, Axiomatizations for some Intuitionistic Modal Logics, in: Rendiconti del Seminario Matematico Università e Politecnico di Torino, 42, 1984, pp. 179–194.

[3] G. D. Plotkin, C. Stirling, A Framework for Intuitionistic Modal Logics, in: Proceedings of the 1st Conference on Theoretical Aspects of Reasoning about Knowledge, Monterey, CA, USA, March 1986, 1986, pp. 399–406.

[4] T. Braüner, V. de Paiva, Intuitionistic Hybrid Logic, Journal of Applied Logic 4 (2006) 231–255. doi:10.1016/j.jal.2005.06.009.

[5] B. Alkmim, E. H. Haeusler, D. Schwabe, A Case Study Integrating Knowledge Graphs and Intuitionistic Logic, in: V. Rodríguez-Doncel, M. Palmirani, M. Araszkiewicz, P. Casanovas, U. Pagallo, G. Sartor (Eds.), AI Approaches to the Complexity of Legal Systems XI-XII - AICOL International Workshops 2018 and 2020: AICOL-XI@JURIX 2018, AICOL-XII@JURIX 2020, XAILA@JURIX 2020, Revised Selected Papers, volume 13048 of *Lecture Notes in Computer Science*, Springer, 2020, pp. 106–124. URL: https://doi.org/10.1007/978-3-030-89811-3_8. doi:10.1007/978-3-030-89811-3_8.

[6] K. Menger, A Logic of the Doubtful: On Optative and Imperative Logic, Reports of a Mathematical Colloquium (2nd series, 2nd issue) (1939) 53—-64.

[7] G.-J. C. Lokhorst, An Intuitionistic Reformulation of Mally's Deontic Logic, Journal of Philosophical Logic 42 (2013) 635–641. URL: http://www.jstor.org/stable/42001179.

[8] R. Goldblatt, Cover Semantics for Quantified Lax Logic, J. Log. and Comput. 21 (2011) 1035–1063. URL: https://doi.org/10.1093/logcom/exq029. doi:10.1093/logcom/exq029.

[9] M. Gavanelli, E. Lamma, F. Riguzzi, E. Bellodi, R. Zese, G. Cota, Abductive Logic Programming for Normative Reasoning and Ontologies, in: New Frontiers in Artificial Intelligence - JSAI-isAI 2015 Workshops, LENLS, JURISIN, AAA, HAT-MASH, TSDAA, ASD-HR, and SKL, Kanagawa, Japan, November 16-18, 2015, Revised Selected Papers, 2015, pp. 187–203. URL: https://doi.org/10.1007/978-3-319-50953-2_14. doi:10.1007/978-3-319-50953-2_14.

[10] E. H. Haeusler, A. Rademaker, On How Kelsenian Jurisprudence and Intuitionistic Logic Help to Avoid Contrary-to-Duty Paradoxes in Legal Ontologies, in: Lógica no Avião Seminars, volume 1, Univeristy of Brasília, 2019, pp. 44–59. URL: http://doi.org/c768.

[11] B. Alkmim, E. H. Haeusler, A. Rademaker, Utilizing iALC to Formalize the Brazilian OAB Exam, in: G. J. Nalepa, M. Atzmueller, M. Araszkiewicz, P. Novais (Eds.), Proceedings of the EXplainable AI in Law Workshop co-located with the 31st International Conference on Legal Knowledge and Information Systems, XAILA@JURIX 2018, Groningen, The Netherlands, December 12, 2018, volume 2381 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2018, pp. 42–50. URL: http://ceur-ws.org/Vol-2381/xaila2018_paper_2.pdf.

[12] B. Alkmim, A Lógica Sobre Leis iALC: Implementação de Provas de Correção e Completude e Proposta de Formalização da Legislação Brasileira (in Portuguese), Masters thesis, Pontifícia Universidade Católica do Rio de Janeiro, 2019.

[13] K. Satoh, K. Asai, T. Kogawa, M. Kubota, M. Nakamura, Y. Nishigai, K. Shirakawa, C. Takano, PROLEG: An Implementation of the Presupposed Ultimate Fact Theory of Japanese Civil

Code by PROLOG Technology, in: T. Onada, D. Bekki, E. McCready (Eds.), New Frontiers in Artificial Intelligence, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 153–164.

[14] M. J. Sergot, F. Sadri, R. A. Kowalski, F. Kriwaczek, P. Hammond, H. T. Cory, The British Nationality Act as a Logic Program, Commun. ACM 29 (1986) 370–386. URL: https://doi.org/10.1145/5689.5920. doi:10.1145/5689.5920.

[15] M. Ferrari, C. Fiorentini, G. Fiorino, BCDL: Basic Constructive Description Logic, Journal of Automated Reasoning 44 (2010) 371–399.

[16] V. de Paiva, E. H. Haeusler, A. Rademaker, Constructive Description Logics Hybrid-Style, Electronic Notes in Theoretical Computer Science 273 (2011) 21–31. URL: https://doi.org/10.1016/j.entcs.2011.06.010, International Workshop on Hybrid Logic and Applications 2010 (HyLo 2010).

[17] E. H. Haeusler, V. d. Paiva, A. Rademaker, Intuitionistic Description Logic and Legal Reasoning, in: 2011 22nd International Workshop on Database and Expert Systems Applications, 2011, pp. 345–349. URL: https://doi.org/10.1109/DEXA.2011.46.

[18] P. Delfino, B. Cuconato, E. H. Haeusler, A. Rademaker, Passing the Brazilian OAB Exam: Data Preparation and Some Experiments, in: A. Z. Wyner, G. Casini (Eds.), Legal Knowledge and Information Systems - JURIX 2017: The Thirtieth Annual Conference, Luxembourg, 13-15 December 2017, volume 302 of Frontiers in Artificial Intelligence and Applications, IOS Press, 2017, pp. 89–94. URL: https://doi.org/10.3233/978-1-61499-838-9-89.

[19] A. Rademaker, A Proof Theory for Description Logics, Springer Publishing Company, Incorporated, 2012.

[20] E. Haeusler, V. De Paiva, A. Rademaker, Intuitionistic Logic and Legal Ontologies, in: Legal Knowledge and Information Systems - JURIX 2010, volume 223 of Frontiers in Artificial Intelligence and Applications, IOS Press, 2010, pp. 155–158. URL: https://doi.org/10.3233/978-1-60750-682-9-155.

[21] D. Prawitz, Natural Deduction: a Proof-theoretical Study, Ph.D. thesis, Almqvist & Wiksell, 1965. URL: https://doi.org/10.2307/2271676.

[22] L. C. Pereira, R. O. Rodriguez, Normalization, Soundness and Completeness for the Propositional Fragment of Prawitz' Ecumenical System, Revista Portuguesa de Filosofia 73 (2017) 1153–1168. URL: http://www.jstor.org/stable/26291332.

[23] Y. Andou, A Normalization Procedure for the First Order Classical Natural Deduction with Full Logical Symbols, Tsukuba Journal of Mathematics 19 (1995) 153–162. URL: http://www.jstor.org/stable/43685916.

[24] G. Pottinger, A New Way of Normalizing Intuitionist Propositional Logic, Studia Logica 35 (1976) 387–408. doi:10.1007/BF02123405.

[25] M. Mendler, S. Scheele, Towards Constructive DL for Abstraction and Refinement, Journal of Automated Reasoning 44 (2009) 207–243.

[26] E. H. Haeusler, M. Benevides, V. de Paiva, A. Rademaker, On the Computational Complexity of the Intuitionistic Hybrid Modal Logics, in: Proceedings of 17th Brazilian Logic Conference, 2014, p. 54.