Proof Translations for Intuitionistic Modal Logic

MSc Thesis *(Afstudeerscriptie)*

written by

Justus Becker

under the supervision of **Marianna Girlando** and **Yde Venema**, and submitted to the Examinations Board in partial fulfillment of the requirements for the degree of

MSc in Logic

at the *Universiteit van Amsterdam.*

Date of the public defense:
26 August, 2024

Members of the Thesis Committee:
Dr Marianna Girlando (supervisor)
Prof Dr Yde Venema (co-supervisor)
Prof Dr Benno van den Berg (chair)
Prof Dr Revantha Ramanayake

INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

Contents

Abstract

Extending sequent calculus with further structure or language has been a fruitful approach in structural proof theory for finding different cut-free complete proof systems. Finding translations between these different calculi not only gives new completeness results, but also insights into how these different calculi compare. While most research has gone into comparing systems that can be seen as notational variants of each other, only some investigation has been done into systems that are structurally different. Two such structurally different calculi exist for the intuitionistic modal logic IK, namely the (simple) nested Maehara-style calculus, NIKm, and the bi-labelled semantic calculus, $\text{labIK}_{\leq}$. Intuitionistic modal logics are modal logics whose modal free fragments are an intuitionistic (propositional or predicate) logic, one of which is IK which is defined in the tradition of Fischer Servi. In this thesis we will introduce intuitionistic modal logics together with their Kripke semantics and some of its proof theory, before establishing an effective two way translation between $\text{labIK}_{\leq}$ and NIKm. Constructing the translations relies on admissible and derivable rules in these systems as well as constructing translatable derivations. The translations that we define shed new light on the relationship between invertibility and back-tracking in proof search, including how specifically these two systems compare to each other.

Acknowledgements

First and foremost, I want to thank my supervisor, Marianna, for guiding me through the process of writing this thesis. Our almost weekly meetings were always very insightful and inspired me to learn more about proof theory of intuitionistic and modal logic. Discussing (and occasionally disagreeing) about the details of proofs, especially, helped me develop my mathematical research skill set further.

I also want to thank the other members of the thesis committee for taking the time and effort to read my thesis. Your questions helped me view my work from a fresh perspective and your comments were encouraging such that the defence felt more like an informal presentation.

Finally, I want to thank my friends and family for their support, especially my partner, Emma, who proofread this thesis for typos and punctuation.

Chapter 1

Introduction

The approach of structural proof theory to obtain more expressive proof systems by extending the sequent formalism has been quite influential, especially for the proof theory of modal logic. Over the last decades, this allowed proof theorists to implement innovative proof systems for various logics. It also allowed for the introduction of calculi for logics for which no previous system existed; for example, there exist cut-free complete systems for $\mathsf{S5}$, but none of which use pure sequents. Moreover, it also allows one to obtain alternative systems for the same logic which have different properties than their original sequent calculus; consider for instance Fitting's [21] and Gentzen's [22] calculus for IPL. Extending the structure or the language to gain a more expressive proof system also enables one to define modular systems that can handle multiple logics at once by adding or removing rules (e.g. modal logics of the $\mathsf{S5}$ cube or intermediate logics).

Besides there being multiple styles of proof systems (for instance, natural deduction, tableaux, or sequent calculus), there are also often different systems of the same style that are sound and complete with respect to the same logic. We can, for instance find at least seven different pure sequent systems for intuitionistic propositional logic (G1i−G5i, m-G3i and GKi, see [53]). This multitude of systems is further exacerbated by the fact that we can also find different ways of extending sequent calculus (for instance, hypersequents, nested sequents and labelled sequents).

As a tool for comparing these different proof systems, we can use effective translations, which are algorithmically definable functions that transform proofs of one system into valid proofs of another one. This lets us, for example, show completeness in a new way, and also compare systems of different formalisms, like the tree-labelled and tree-hypersequent (or nested) formalism (see [11, 25]). It can also let us see how two different systems behave and transfer structural results that are already established in one system to the other.

We will consider here the proof theory of the intuitionistic modal logic IK, which has been established and developed in the 1980s and 90s as the intuitionistic variant of the logic K (see [18, 17, 45, 50]. Since then, a lot of systems have been developed for the logic.

In this thesis, we will establish two directions of a translation between two fundamentally different calculi for IK; namely, the semantic calculus, $\mathrm{labIK}_{\leq}$ (see [36]), and the simple nested calculus, NIKm (see [31]). These two systems, although they are both based on sequent calculus, are fundamentally different because they are not mere notational variants of each other. While $\mathrm{labIK}_{\leq}$ internalises the bi-relational structure of intuitionistic modal Kripke models, NIKm uses nestings for modalities and gains its intuitionistic behaviour directly from its rules. Further, while $\mathrm{labIK}_{\leq}$

is fully invertible, meaning that the order of rule applications generally does not matter, there are rules in NIKm which are not invertible. This makes the translation significantly more challenging than other translations. One issue is that not every proof is immediately translatable, so the translation will also include an effective algorithm for reaching translatable proofs.

We begin this thesis by introducing the axiomatisation and semantics for IK together with extensions of Horn-Scott-Lemmon axioms. We also discuss why IK can be considered to be the intuitionistic variant of the classical modal logic K. After that, in chapter 3, we introduce the proof theory of the logic by first considering three proof systems for intuitionistic propositional logic. These systems include two pure sequent calculi and a semantic labelled system. Then, we extend these systems towards two proof systems for IK and discuss some of their properties. Finally, in chapter 4, we establish a translation between these two systems by using admissible rules and a construction of derivations.

Chapter 2

Intuitionistic Modal Logic

This chapter introduces the basic notions and definitions of intuitionistic modal logic. This includes the axiomatisation and semantics of the modal logic IK together with extensions, especially extensions with Horn-Scott-Lemmon axioms. We will also discuss a few different variants of intuitionistic modal logic that historically precede IK, though most of them are still relevant nowadays, and discuss shortly why IK can be seen as a good intuitionistic modal logic.

2.1 Preliminaries

Intuitionistic modal logics (IMLs) are generally those modal logics which are based on some intuitionistic logic. This includes usually intuitionistic propositional logic (IPL) or predicate logic (e.g. intuitionistic first-order logic, see [30]). This means that, when taking the modal-free fragment of an IML, one is always left with an intuitionistic logic. Due to their very high complexity, we will not consider modal logics based on predicate logic.

Similar to classical modal logic, we might characterise the language of IML by adding some modal operator to the propositional language of intuitionistic logic (e.g. $\Box$) and define the other one as its dual (e.g. $\Diamond := \neg\Box\neg$). However, because we do not have double negation elimination, the so-defined modality will not be the dual of the primitive modality. Thus, the logic differs depending on which modality we choose as primitive unless we explicitly demand that they are dual to each other (e.g. adding $\neg\Diamond\neg A \supset \Box A$ as an explicit axiom). To keep the dependency open, we define both modalities as primitives. Furthermore, we will see that it might be desirable for an IML to have independent modalities.

Thus, we define our language of intuitionistic modal logic, which we will call $\mathcal{L}^{\Box\Diamond}$, by the following BNF grammar with a countable set of propositional atoms $\Phi = \{p, q, r, ...\}$ and the constant symbol $\bot$ for the always false sentence.

$$A ::= \bot \mid p \mid A \wedge B \mid A \vee B \mid A \supset B \mid \Box A \mid \Diamond A$$

We define the usual abbreviations

$$\neg A := A \supset \bot \text{ and } \top := \neg\bot,$$

as well as $\Box^n A$ and $\Diamond^n A$ for the formula A prefixed with a sequence of n boxes and diamonds respectively (whereas $\Box^0 A = \Diamond^0 A = A$). Call the set of modal free formulas $\mathcal{L}$. We distinguish intuitionistic implication from the classical one, by using $\supset$ instead of $\rightarrow$. We define (uniform) formula substitution in the usual way.

Definition 2.1 (Formula Substitution)**.** *For a formula A from our language $\mathcal{L}^{\Box\Diamond}$, we define the substitution instance $A_{[B/p]}$ with B also being in the language and $p \in \Phi$ by replacing every occurrence of p in A with the formula B. If p does not occur in A, we have $A_{[B/p]} = A$.*

The condition that the modal-free fragment of the basic IML reduces to IPL still leaves a lot of freedom for choosing a "basic" IML. Here, the question is not so much on what is the one correct IML, but rather what is an intuitionistic modal logic which has generally good properties such that it can be considered a basic intuitionistic logic as well as the smallest normal modal logic. A first natural candidate for such a role is to consider the $\Diamond$-free fragment of the modal language together with the well-known axiom $\Box(A \supset B) \supset (\Box A \supset \Box B)$ (K) which also defines the "basic" classical modal logic. This logic was introduced by Božić and Došen in 1984 and is now known as iK (see [7]). To actually introduce a full modal logic by also considering $\Diamond$ as an active connective, one can extend this system with further axioms that give it a more and more *normal* behaviour (as in *normal modal logic*).[1] Furthermore, we would want $\Box$ and $\Diamond$ to have at least some similarity in their classical behaviour. The system one might get from that is the logic IK which was introduced in the 1980s independently under three different axiomatisations ([16, 17, 45]).

In his PhD thesis, Alex Simpson discusses this question of what is (a good) intuitionistic modal logic (see [50, Chapter 3.2]). The most important point he makes, which we also already mentioned, is that any IML should be conservative over IPL. Another point we already touched upon is the independence of the two modalities, $\Box$ and $\Diamond$, which Simpson also considers to be an important feature of intuitionistic modal logic. The main reason being that, just like the other classical duals (quantifiers and disjunction-conjunction), the modalities should be independent in intuitonistic logic. This can also be supported by requiring that there should be some intuitionistic meaning for the modalities. Simpson formalises this broad demand for intuitionistic meaning into, firstly, requiring that the standard translation of an IML should be some fragment of intuitionistic first-order logic and, secondly, by a natural deduction system that incorporates both aspects of how the modalities usually behave and of how natural deduction for IPL works (no classical *reductio ad absurdum* rule). Additionally, we would at least expect that an IML has the disjunction property and would become a classical modal logic when adding the law of excluded middle. As Simpson shows, all these conditions apply to IK and therefore support it as a good candidate for a "basic" IML.

2.2 Hilbert-Axiomatisation for Intuitionistic Modal Logic

Let us begin by introducing the aforementioned intuitionistic modal logic IK, standing for *intuitionistic* K, which is defined by means of a Hilbert system as follows.

[1] One property of the (classical) non-normal modal logic E is that it falsifies the distributivity of $\Diamond$ over disjunction ($\Diamond(A \vee B) \supset (\Diamond A \vee \Diamond B)$). Classically this axiom follows from K, whereas it has to be added explicitly for intuitionistic modal logic.

Definition 2.2 (Syntactic Axiomatisation)**.** *The Hilbert-style axiomatisation of* IK *is defined by the following axioms:*

All intuitionistic propositional tautologies (IPL)

$k_1 : \Box(A \supset B) \supset (\Box A \supset \Box B)$

$k_2 : \Box(A \supset B) \supset (\Diamond A \supset \Diamond B)$

$k_3 : \Diamond(A \vee B) \supset (\Diamond A \vee \Diamond B)$

$k_4 : (\Diamond A \supset \Box B) \supset \Box(A \supset B)$

$k_5 : \neg\Diamond\bot$

And it is closed under the following rules:

(mp) From $A \supset B$ *and* A *infer* B

(nec) From A *infer* $\Box A$

If a formula A *is provable in* IK*, we write* $\mathsf{IK} \vdash A$*. If* $A \in \mathcal{L}$ *we can also write* $\mathsf{IPL} \vdash A$*.*

For a Hilbert-style axiomatisation of intuitionistic logic, see [53, Definition 2.4.1], where IPL is defined as the purely propositional part.

The logic consisting only of the additional modal axiom k_1 is called iK, which we already mentioned as the $\Diamond$-free fragment of some intuitionistic modal logic. We note, however, that this does not imply that all intuitionistic modal logics between iK and IK prove the same $\Box$-only formulas, as it even has been shown that the $\Diamond$-free fragments of iK and IK are distinct (see [12]). Adding k_2 to iK and closing it under the given rules gives us the *constructive* modal logic CK. A first-order variant of it was introduced in [19] as the very first intuitionistic modal logic. Further, adding k_5 gives us the propositional fragment of the Constructive Concurrent Dynamic Logic (CCDL) from Wijesekera (see [56]). The main difference between these systems and IK is that, as precursors, they were mostly motivated by applications to computer science, including, for example, Curry-Howard correspondence (see also [3]). This is also why those logics still have a strong relevance today, though we will not touch upon them here any further.

Similar to how we can extend classical modal logic with further modal axioms, as well as intuitionistic propositional logic with formulas to gain superintuitionistic logics (also known as *intermediate logics*), we can do both for IML as well. We call those extensions, which form well behaved theories: *intuitionistic normal modal logics.*

Definition 2.3 (Intuitionistic Normal Modal Logic)**.** *An intuitionistic normal modal logic* L *contains formulas of the language* $\mathcal{L}^{\Box\Diamond}$ *such that* $\mathsf{L} \supseteq \mathsf{IK}$ *and* L *is closed under mp, nec and formula substitution (from* A *infer* $A_{[B/p]}$ *for* $B \in \mathcal{L}^{\Box\Diamond}$*).*

Remark 2.4. When extending these logics to classical ones by adding the excluded middle principle, we obtain a (classical) *normal modal logic* (see [6, Definition 1.42]), thus our naming is also sensible.

Similar to how there are a lot of axioms which are classically equivalent to K, but which one has to add explicitly for IK, the same phenomenon applies to modal extension axioms. For example, when classically adding axiom 4 to the logic, it does not matter whether it is written as $\Box A \supset \Box\Box A$ or $\Diamond\Diamond A \supset \Diamond A$. For IMLs, these formulas are nonequivalent. We include both formulations as extension axioms. The axiom schemas for the S5-cube extensions for intuitionistic modal logic are then defined as in the following table (figure 3.3.3).

	Axiom
D	$\Box A \supset \Diamond A$
B	$(\Diamond\Box A \supset A) \wedge (A \supset \Box\Diamond A)$
T	$(\Box A \supset A) \wedge (A \supset \Diamond A)$
4	$(\Box A \supset \Box\Box A) \wedge (\Diamond\Diamond A \supset \Diamond A)$
5	$(\Diamond A \supset \Box\Diamond A) \wedge (\Diamond\Box A \supset \Box A)$

Figure 2.2.1: Extension axioms for logics in the IS5 cube

The full set of logics that we can gain from adding any subset of these axioms to IK is represented in figure 2.2.2. These logics between IK and IS5 constitute the so called "classical" cube or simply IS5 cube.

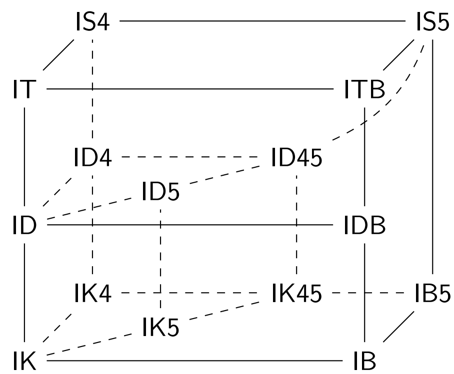


Figure 2.2.2: The IS5 cube

2.3 Semantics for Intuitionistic Modal Logic

Similar to how one can use Kripke models to either describe classical modal logics or for modelling IPL and their superintuitionistic extensions, one can combine these notions into a single structure. The idea was motivated by Fischer Servi (see [18]) by considering the Gödel translation of IPL into S4, transforming IML into a bimodal logic.

Definition 2.5. *A birelational model is a tuple $\langle W, \leq, R, V \rangle$ with a non-empty set W, a reflexive and transitive relation $\leq\, \subseteq W \times W$, a relation $R \subseteq W \times W$ and a valuation function $V : \Phi \to \mathcal{P}(W)$ with the following conditions.*
Monotonicity: If $x \in V(p)$ and $x \leq y$ then $y \in V(p)$.
(F1): If xRy and $y \leq z$ then there is some z' such that $z'Rz$ and $x \leq z'$.
(F2): If xRy and $x \leq z$ then there is some z' such that zRz' and $y \leq z'$.

Another way the monotonicity condition is often written down is by demanding that V be a monotone mapping from $(W, \leq)$ to $(2^{\Phi}, \subseteq)$. This requires one to connect states and atomic propositions to binary truth values for the truth condition. Under this distinction, both definitions are equivalent. The conditions (F1) and (F2) are also known respectively as the *backward* and *forward confluence* condition (also see figure 2.3.1). While (F2) is necessary to obtain general monotonicity (see lemma 2.8), (F1) helps make true axiom k_4 of IK.[2] We note that there is no coherent labelling throughout the literature for the confluence rules; though most papers use the same notation as here, whereas Simpson (see [50]) has the labels swapped (i.e. forward confluence is named (F1) and backward is (F2)).

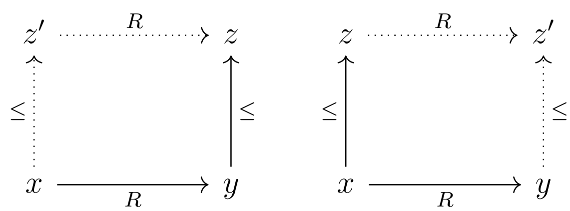


Figure 2.3.1: Backward and forward confluence condition

When talking about the relations of a birelational model, we define for any natural number $n \geq 0$ the relation xR^ny to mean that y is reachable in exactly n steps via R, whereas xR^0y means $x = y$.

Truth in the semantics is defined in the same way as for either modal and intuitionistic Kripke models, except for the modality $\Box$, whose truth relies on both relations ($\leq$ and R) and is equivalent to $\top \supset \blacksquare A$ for a classical modality $\blacksquare$. This definition of the $\Box$ modality also enables monotonicity to work.

Definition 2.6 (Truth in a state)**.** *The forcing relation $\Vdash$ on a state w in a model $\mathcal{M}$ is defined inductively on the complexity of formulas:*
$\mathcal{M}, w \Vdash \bot$ *never*
$\mathcal{M}, w \Vdash p$ *iff* $w \in V(p)$
$\mathcal{M}, w \Vdash A \wedge B$ *iff* $\mathcal{M}, w \Vdash A$ *and* $\mathcal{M}, w \Vdash B$
$\mathcal{M}, w \Vdash A \vee B$ *iff* $\mathcal{M}, w \Vdash A$ *or* $\mathcal{M}, w \Vdash B$
$\mathcal{M}, w \Vdash A \supset B$ *iff for all* $v \in W$*: if* $w \leq v$ *and* $\mathcal{M}, v \Vdash A$ *then* $\mathcal{M}, v \Vdash B$
$\mathcal{M}, w \Vdash \Box A$ *iff for all* $v, u \in W$*: if* $w \leq v$ *and* vRu *then* $\mathcal{M}, u \Vdash A$
$\mathcal{M}, w \Vdash \Diamond A$ *iff there is a* $v \in W$ *such that* wRv *and* $\mathcal{M}, v \Vdash A$

We say that a sentence A is *true in a model* $\mathcal{M}$ (written $\mathcal{M} \models A$) iff $\mathcal{M}, w \Vdash A$ for all states w in the model. As usual, a *frame* $\langle W, \leq, R \rangle$ can be seen as a model without valuation. *Truth in a frame* is defined as truth in any model based on that frame. As usual, $\nVdash$ means that the state, the model, or the frame does not make that formula valid.

Remark 2.7. If the relation R is empty for a bimodal model, definitions 2.5 and 2.6 give the exact definition of a Kripke model for IPL. Similarly, if $\leq$ is trivial (only reflexive relations) the model constitutes a Kripke model for the classical modal logic K.

Just like in intuitionistic Kripke models, in bimodal models the excluded middle principle is invalidated by the negation relying on $\leq$. At the same time, we also have the following lemma, reminiscent of the monotonicity condition for propositional atoms.

[2]Note that removing (F1) will not be equivalent to removing k_4. See also [2].

Lemma 2.8 (Monotonicity)**.** *If $\mathcal{M}, w \Vdash A$ and $w \leq v$ then $\mathcal{M}, v \Vdash A$ for $A \in \mathcal{L}^{\Box\Diamond}$.*

Proof. By induction on the complexity of A. The base case is just monotonicity of the valuation. Non-trivial cases are $A \supset B$ and $\Box A$ by using transitivity of $\leq$ and $\Diamond A$ by using (F2). □

Just like for classical modal logic, there is a correspondence between modal extensions of IK and frame conditions on the relations of the models. On the one hand, many well-known formula extensions of the classical modal logic K correspond nicely to a first-order property of the R-relation on frames. It is, on the other hand, a well known fact that there are many formula extensions in modal logic that do only correspond to second-order definable properties; for example, the Löb axiom $\Box(\Box A \rightarrow A) \rightarrow \Box A$ corresponds to frames that are conversely well-founded. Additionally, some classical modal logics do not even correspond to any class of frames (frame incompleteness) and are only complete wrt. general frames.

For these reasons, we only consider a small and well behaved class of modal axioms: the so-called Horn-Scott-Lemmon axioms (HSLs). They form a proper subclass of Scott-Lemmon axioms and therefore have also a simple first-order correspondence. We consider here their two-sided versions (meaning that we include both duals), which are classically equivalent. They are defined by the following schema with $n, m \in \mathbb{N}$.

$$\phi(n, m) := (\Diamond^n \Box A \supset \Box^m A) \wedge (\Diamond^m A \supset \Box^n \Diamond A)$$

As we will see, adding such a formula as an axiom to IK is equivalent to restricting the class of frames under the condition: If $xR^n y$ and $xR^m z$ then yRz. We refer to the set of all Horn-Scott-Lemmon axioms simply as HSL. Notice that we make no usage of the $\leq$-relation, which is mainly due to us using both classically equivalent versions of one-sided HSLs.

HSLs include many of the well-known modal formulas that have a first-order frame correspondence, e.g. the axioms B, T, 4 and 5. We additionally consider the directedness axiom $\Box A \supset \Diamond A$ (D) with the seriality condition $\forall x \exists y(xRy)$ (equivalently one can use $\Diamond\top$ as the axiom D). This allows us to model all logics inside the classical cube. We refer to a subset of these axioms as $\mathcal{A} \subseteq \{D\} \cup \text{HSL}$. The logic $\mathsf{IK} \oplus \mathcal{A}$ is then defined as the logic IK together with the formulas from $\mathcal{A}$ closed under necessitation and modus ponens. Because we are always introducing new axioms as full schemata, these logics will also obey formula substitution and are therefore intuitionistic normal modal logics. For the extensions in the IS5 cube, we use the better known names for these logics, like in figure 2.2.2; e.g. $\mathsf{IS4} = \mathsf{IK} \oplus \{\phi(0,0), \phi(0,2)\}$ and $\mathsf{IKD5} = \mathsf{IK} \oplus \{D, \phi(1,1)\}$.

To state completeness of the class of modal logics which are characterised by $\mathsf{IK} \oplus \mathcal{A}$, we define models that satisfy the corresponding frame conditions.

Definition 2.9 ($\mathcal{A}$-model)**.** *Let $\mathcal{A} \subseteq \{D\} \cup HSL$ be a set of formulas. We call a model $\mathcal{M}$ an $\mathcal{A}$-model, if the following conditions are met:*

1. *$D \in \mathcal{A}$: For all x there is some y s.t. xRy.*
2. *$\phi(n, m) \in \mathcal{A}$: If $xR^n y$ and $xR^m z$ then yRz, for any x, y, z in $\mathcal{M}$.*

As a side note, we can reduce the frame conditions if at least one of the indices is zero, which will also give us the usual formulation of frame conditions for frames of logic in the S5 cube (e.g. reflexivity, transitivity and symmetry). This reformulation is simply done by considering that $xR^0 y$ means $x = y$ and therefore simply replacing one variable for another.

This lets us formulate the most simple completeness for an infinite collection of logics.

Axiom	Frame condition
$\phi(0,0)$	xRx
$\phi(n,0)$	If $xR^n y$ then yRx
$\phi(0,m)$	If $xR^m y$ then xRy

Figure 2.3.2: Frame conditions for HSLs with at least one index being zero

Theorem 2.10 (Soundness and Completeness)**.** *Let $\mathcal{A} \subseteq \{D\} \cup HSL$ be a set of formulas, then the logic $\mathsf{IK} \oplus \mathcal{A}$ is sound and complete wrt. the class of $\mathcal{A}$-models, meaning that for any formula A: $\mathsf{IK} \oplus \mathcal{A} \vdash A$ iff $\mathcal{M} \Vdash A$ for all $\mathcal{A}$-models $\mathcal{M}$.*

Proof. Follows from Theorem 6.2.1 and Theorem 8.1.4 of [50]. □

An earlier completeness result had been obtained by Fischer Servi ten years earlier for some of the intuitionistic modal logics in the classical cube [17]. A stronger result was also established earlier in [45], where the authors consider a frame correspondence for Scott-Lemmon axioms instead of its proper subset HSL. This correspondence, though, relies on a frame condition that uses both relation of the frame and was not proved as it is only part of an abstract.

Chapter 3

Proof Theory of Intuitionistic Modal Logic

In this chapter, we will present a multitude of calculi for intuitionistic propositional logic and calculi for intuitionistic modal logic. As it is common practice in modern proof theory, we will solely consider calculi based on Gentzen's sequent systems due to them being easy to analyse and obeying proof search procedures that are often simpler than those of natural deduction or tableaux systems. At first, we will look at three important proof systems for intuitionistic propositional logic, which get introduced in section 3.2. These are the single succedent calculus introduced by Gentzen, the Maehara-style multi succedent calculus and a labelled system. They will give us a vital basis for understanding the systems in which we also handle modalities. In section 3.3, we consider calculi for the previously introduced logic IK together with extensions of the Horn-Scott-Lemmon axioms.

3.1 Sequent Calculus

One of the most important types of calculi for proof theorists are so-called Gentzen systems. They were originally introduced by Gentzen in 1935 (see [22]) and subsequently named after him. Most modern *G-systems* are build on his initial calculi for classical and intuitionistic first-order logic, which he called LK and LI (nowadays also known as G1c and G1i, denoting *the first* Gentzen calculi).[1] While *K* stands for *klassisch* (German for *classical*) and *I* for *intuitionistic*, the *L* in LI and LK is an abbreviation for *logistic*, expressing the fact that these calculi do not rely on the dependency of global assumptions. Contrastingly, natural deduction, for example, does often heavily rely on temporary assumptions that are only later lifted at a certain point in the derivation process.

Unlike other deductive systems, Gentzen calculi work on sequents instead of directly on formulas. These are usually of the form $\Gamma \Rightarrow \Delta$, where both Γ and Δ are multisets of formulas (i.e. sets

[1] Often in papers and books, one reads *LJ* instead of *LI*. This is mainly due to Gentzen using the Sütterlin handwriting for certain formulations, in which "*I*" looks like "$\mathfrak{J}$" (see [44, p.83]). Even in the original 1935 print version of Genzten's *Untersuchungen über das logische Schließen (Investigations into Logical Deduction)* the Sütterlin *I* is printed as a *J*, which might have actually kick-started a lot of the mislabelling.

which may contain some elements multiple times, or lists without any order).[2] These multisets are distinguished by an arrow $\Rightarrow$, where formulas on the left side are part of the *antecedent* while the one on the right are called *succedent* formulas. In classical logic, such a sequent can be interpreted as a formula in the language:

$$fm(\Gamma \Rightarrow \Delta) = \bigwedge_{A \in \Gamma} A \to \bigvee_{B \in \Delta} B$$

Such an interpretation also works for certain sequents in the intuitionistic case. This gives sequents a very natural reading.

Making derivations on sequents, rather than on formulas, grants the sequent calculus the ability to work locally. These rules are local in the sense that we do not have to look for other formulas in the derivation except for the formulas inside the antecedent and succedent. The rules have thus the simple form

$$\frac{\mathfrak{S}_1, ..., \mathfrak{S}_n}{\mathfrak{C}}$$

with $n = 0, 1, 2$ and $\mathfrak{S}_1, ..., \mathfrak{S}_n, \mathfrak{C}$ referring to sequents. This allows us to define valid derivations for any sequent system, which are also called proof trees or derivation trees. We will generally write $\mathfrak{S}$ to refer to two-sided sequents. A collection of rules is called a *(proof) system* or *calculus*.

Definition 3.1 (Derivation Tree)**.** *A proof tree $\mathcal{T}$ is a rooted tree whose nodes are sequents which are connected via rule applications. We say that a proof tree is valid if every sequent has a number of ancestors (0, 1 or 2) such that their syntax matches the form of one of the rules. Otherwise, the tree is invalid.*

We say a sequent $\mathfrak{S}$ is provable in a system L, written as $L \vdash \mathfrak{S}$, iff there is a valid proof tree of L with $\mathfrak{S}$ as the root of the tree. A formula A is provable in L, written $L \vdash A$ iff $L \vdash (\Rightarrow A)$.

If a proof tree is invalid, we might distinguish two main cases: either, it has a mistake and there is a connection between sequents that does not follow any possible rule application, or there are sequents which is not introduced by a rule (not even a zero premise rule). These latter sequents are called *open leafs*, as they might still be extendable in the derivation tree.

Initially, rule applications in a derivation tree were being thought of as a top-down procedure to derive some formula from a set of chosen premises. Nowadays, the idea of rule application in Gentzen calculus is closer to how the tableaux systems are being used, where one starts out with a formula one wants to prove (or disprove) and does a proof search procedure. This gives us the two notions of applying rules: *upwards* rule application always start with the conclusion, like in a proof search, whereas *downwards* applications begin with axioms or premises and go to the conclusion.

We define the *height of a proof tree* $h(\mathcal{T})$ as the length of the longest branch of the tree from the root to a leaf. If the derivation of a formula A is at least of height n, we say that A is n-*provable* ($\vdash_n A$). One can also say that the formula is provable in n many steps, though that can be misleading as there might be more steps (i.e. rule applications) due to branching.

[2]There are, especially in the more foundational texts, versions of sequent calculus which use lists or sets of formulas. These formulations are usually equivalent.

Besides the already mentioned feature of being local, sequent calculi are also more often than not analytic, meaning that, as we progress in the proof tree, formulas become less and less complex. This makes them an elegant and versatile formalism for proof-theoretic investigation, giving one easy ways to characterise well-behaved systems. One of the features of a well-behaved system is the *admissibility* of certain rules. Admissible rules can be thought of as rules one is "allowed" to make in a system without them being explicitly in the rule set. They are properly defined as follows.

Definition 3.2 (Admissible rules)**.** *A rule, with premises* $\mathfrak{S}_1, ..., \mathfrak{S}_n$ *and a conclusion* $\mathfrak{C}$*, is admissible in a system* L *iff: If* $L \vdash \mathfrak{S}_i$ *for all* i*, then* $L \vdash \mathfrak{C}$*. A rule is height preserving admissible (hp-admissible) iff: if every* $\mathfrak{S}_i$ *is n-provable, then* $\mathfrak{C}$ *is also n-provable.*

The most common admissible rules are called *structural rules* (see e.g. figure 3.2.4); this usually excludes the cut rule, which has a special status among admissible rules. By allowing for cuts in a derivation tree, one can emulate the rule of modus ponens of the Hilbert-style axiomatisation, showing that the system is complete in a syntactic way. In G1 systems, some of the structural rules occur explicitly and can therefore only be admitted insofar as they are already part of the rule set.

After Gentzen's groundbreaking idea of utilising sequents, consecutive systems have been developed: most notably, systems of G2 and G3, which were developed later to absorb the structural rules into the system, the latter being mainly attributed to works from Ketonen, Kleene and Ono (see [28, 29, 42]). G3 systems are also inspired by Dragalin's calculus (originally published in Russian in 1979, see [13] for a translated reference), which is our system m-G3i (see section 3.2.2); he was also the one to observe that the intuitionistic system G3i can be improved in terms of admissibility of contraction by repeating the principal formula in $\supset$L (see figure 3.2.1). G2 systems were merely invented as didactic stepping stones towards G3 in [53].

A stronger notion of admissibility is *eliminability*, which means that not only can we add such a rule to the whole calculus without changing it, but we can also remove it from a valid derivation in a systematic way. One of the most important elimination results obtainable is the elimination of the cut rule (cut-elimination). The explicit eliminability of cuts is especially important because of its applications like interpolation or explicit definability (see [53, Chapter 4]).

As G3 systems are the most well behaved systems due to their many admissible structural rules, they can be seen as the gold-standard of modern proof systems.

Besides sequent systems, there are also formulations using so-called *one-sided sequents*, first introduced by Schütte in 1950 [49]. Such "sequents" are in principle nothing more than simple multi-sets of formulas, where every formula is a succedent formula. They allow applications for classical calculi due to the usage of De-Morgan's law and double negation elimination, which allows one to write any formula in negation normal form, where the antecedent formulas occur negated while succedent formulas occur as they are.

Intuitionistic calculi can be easily obtained from classical ones by only allowing for a single succedent formula in sequents. Additionally, there are further systems for intuitionistic logic (G4i and G5i) which can keep more formulas in the succedent. This is achieved by having multiple rules for dealing with an implication on the left side of the sequent. An advantage these systems offer is that they are not prone to backtracking; *backtracking* is the moment in proof search where one has to go back in a derivation branch to change a rule application (for a formal definition see section 3.2.3). This might happen, for example, in G3i, when analysing a disjunction in the succedent and only one of the disjuncts leads to a valid derivation. The systems we will consider from now on will be purely propositional and based on the system G3i.

3.2 Intuitionistic Propositional Logic

In this section, we will introduce a multitude of decision procedures for the logic IPL. These include specifically the single succedent calculus, which is simply a restricted version of the classical G3c calculus. We furthermore investigate a variant of intuitionistic sequent calculus where multiple succedent formulas are allowed. Finally, we take a look at a semantic calculus for some intermediate logics, which was introduced by Dyckhoff and Negri [15] which is a notational variant of Fitting's nested calculus (see [21]).

3.2.1 Single Succedent Calculus for IPL

As mentioned before, the usual way of creating an intuitionistic calculus from a classical one is to restrict all sequents to *single succedent sequents* $\Gamma \Rightarrow A$. The rules for the main intuitionistic Gentzen calculus, G3i, are given in figure 3.2.1 (see [53]). We call the set that is named Γ, possibly together with the formula C, the context of the rule, which in our investigations will most often be preserved in rule applications. The formulas that are not in the context, and to which the rules get applied to, are called *principal* in the rule application.

In a way, single succedent[3] calculi model a constructive way of reasoning by the succedent being only dependent on antecedent formulas and never on other succedent formulas.

$$\text{Ax}\frac{}{p, \Gamma \Rightarrow p} \qquad \bot\frac{}{\bot, \Gamma \Rightarrow A}$$

$$\wedge\text{L}\frac{A, B, \Gamma \Rightarrow C}{A \wedge B, \Gamma \Rightarrow C} \qquad \wedge\text{R}\frac{\Gamma \Rightarrow A \qquad \Gamma \Rightarrow B}{\Gamma \Rightarrow A \wedge B}$$

$$\vee\text{L}\frac{A, \Gamma \Rightarrow C \qquad B, \Gamma \Rightarrow C}{A \vee B, \Gamma \Rightarrow C} \qquad \vee\text{R}\frac{\Gamma \Rightarrow A_i}{\Gamma \Rightarrow A_0 \vee A_1}(i \in \{0,1\})$$

$$\supset\text{L}\frac{A \supset B, \Gamma \Rightarrow A \qquad B, \Gamma \Rightarrow C}{A \supset B, \Gamma \Rightarrow C} \qquad \supset\text{R}\frac{A, \Gamma \Rightarrow B}{\Gamma \Rightarrow A \supset B}$$

Figure 3.2.1: Rules for G3i

Special admissible rules, which we can add to a system for free, are derivable rules. We call a rule *derivable* if we can derive the conclusion in a valid proof tree where one takes the premises of the rule as axioms. For example, the generalised axiom rule

$$\text{Ax}^g\frac{}{\Gamma, A \Rightarrow A}$$

[3]Sometimes these systems are also being referred to as single *conclusion* calculi, which is misleading as we often mean the bottom sequent of a rule when referring to a conclusion.

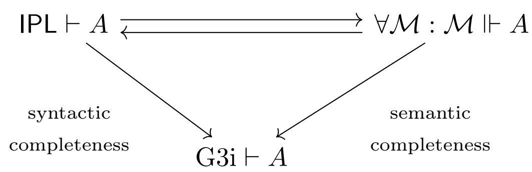


Figure 3.2.2: Two ways of establishing completeness of G3i wrt. IPL

can be derived for any formula A which can be generally proven by induction on the complexity of A. Derivable rules are also eliminable rules, because one can simply replacing the rule with the specific derivation.

The following lemma shows that this system can be considered *well-behaved*.

Lemma 3.3 (Admissible Rules)**.** *The rules* wk*,* sub*,* c_L *and* cut *(figure 3.2.4), restricted to their single succedent form, are admissible in G3i. Furthermore,* wk *and* c_L *are hp-admissible, while* cut *is eliminable.*

Proof. Lemmata 3.5.3, 3.5.5 and theorem 4.1.5 in [53]. For uniform substitution, one can do a simple induction on the depth of the derivation and use the derivable rule Ax^g to substitute for formulas in axiomatic sequents. □

Note that all rules in figure 3.2.4 will generally look different for single succedent sequents. Namely, we always have $\Delta = \emptyset$ and $\Pi = \{C\}$ for some formula C. Also, recall definition 2.1 for formula substitution.

There are usually two ways of proving completeness of a proof system with respect to a logic (see figure 3.2.2). One can prove it semantically, by showing that if $\mathcal{M} \Vdash A$ for any intuitionistic Kripke model $\mathcal{M}$, then G3i $\vdash A$. This is best done by proving the contrapositive, namely if G3i $\nvdash A$ then there is some countermodel $\mathcal{M} \nVdash A$. We will see such a strategy in section 3.2.3.

The other strategy to prove completeness is syntactically by first proving all axioms of the Hilbert system and then using cut-admissibility to show that modus ponens is admissible, thus the system is complete wrt. the Hilbert axiomatisation. This is done by showing that if $\Rightarrow A \supset B$ (implying $A \Rightarrow B$) and $\Rightarrow A$ are derivable, then $\Rightarrow B$ is derivable via the cut rule.

Theorem 3.4 (Soundness and Completeness)**.** *The system G3i is sound and complete wrt. to IPL. Formally we have for any formula $A \in \mathcal{L}$: G3i $\vdash A$ iff* $\mathsf{IPL} \vdash A$*.*

Proof. Follows from theorems 2.4.2, 3.3.1 and proposition 3.5.9 in [53]. □

Remark 3.5. Via the completeness of IPL wrt. their bi-relational models (theorem 2.10) we also have *G3i* $\vdash A$ iff $\mathcal{M} \Vdash A$ for any model $\mathcal{M}$.

Example 3.6 (Failure of excluded middle principle)**.** The formula $p \vee \neg p$, which is classically valid but not intuitionistically, can be written in our language as $p \vee (p \supset \bot)$. Here are the two failed proof trees that would be possible for this formula. They are considered to be failed due to their leaves having no possible further rule application.

$$\dfrac{\Rightarrow p}{\Rightarrow p \vee (p \supset \bot)} \vee R \qquad \dfrac{\dfrac{p \Rightarrow \bot}{\Rightarrow p \supset \bot} \supset R}{\Rightarrow p \vee (p \supset \bot)} \vee R$$

3.2.2 Multi Succedent Sequent Calculus for IPL

Another less radical way to proof-theoretically model intuitionistic propositional logic is to restrict only certain rules of the classical calculus to have single succedent sequents in their premise. One of the most popular variants is obtained by taking the classical calculus and restricting the ⊃R rule to a premise with a single succedent. We call this system m-G3i (see figure 3.2.3) named for the multi conclusion version of G3i, which was first defined by Maehara in [34] as a *Hilfskalkül* (helping calculus), which is closer to the classical system of G3c. This allowed for an easy McKinsey-Tarski-like result of interpreting IPL as S4, essentially modelling intuitionistic logic in classical logic. This is why any intuitionistic calculus with multi succedent sequents is also sometimes referred to as a Maehara-style calculus. Here, the context of the rules is always the sequent $\Gamma \Rightarrow \Delta$. Different formalisms for m-G3i were also independently developed after Maehara, including by Curry [10], and Beth and Fitting [5, 20]; a similar system was also considered by Gentzen himself (see [14, Section 2.2], as well as [44] for a more detailed background).

The reason why m-G3i works as an intuitionistic proof system (and is actually closely related to its historical predecessor) is that the only side changing rule that would allow us to actually utilise multi-succedents is the rule ⊃R.

$$\text{Ax}\dfrac{}{p, \Gamma \Rightarrow \Delta, p} \qquad \bot\dfrac{}{\bot, \Gamma \Rightarrow \Delta}$$

$$\wedge\text{L}\dfrac{A, B, \Gamma \Rightarrow \Delta}{A \wedge B, \Gamma \Rightarrow \Delta} \qquad \wedge\text{R}\dfrac{\Gamma \Rightarrow \Delta, A \qquad \Gamma \Rightarrow \Delta, B}{\Gamma \Rightarrow \Delta, A \wedge B}$$

$$\vee\text{L}\dfrac{A, \Gamma \Rightarrow \Delta \qquad B, \Gamma \Rightarrow \Delta}{A \vee B, \Gamma \Rightarrow \Delta} \qquad \vee\text{R}\dfrac{\Gamma \Rightarrow \Delta, A, B}{\Gamma \Rightarrow \Delta, A \vee B}$$

$$\supset\text{L}\dfrac{A \supset B, \Gamma \Rightarrow \Delta, A \qquad B, \Gamma \Rightarrow \Delta}{A \supset B, \Gamma \Rightarrow \Delta} \qquad \supset\text{R}\dfrac{A, \Gamma \Rightarrow B}{\Gamma \Rightarrow \Delta, A \supset B}$$

Figure 3.2.3: Rules for m-G3i

When considering a derivation in G3i, one can easily translate it into a m-G3i derivation by weakening the respective rule applications of ∨R or ⊃L. The other direction was already obtained by Maehara to gain completeness for his new system in [34]. It was achieved in two steps: Any sequent of m-G3i can be transformed into a single-succedent by forming a disjunction of all succedent formulas, and, in a second step, one can show the admissibility of all rules under this sequent translation.

Due to this strong similarity, it is no wonder that this system enjoys the same structural properties as G3i. So, Lemma 3.3 also applies to it, as well as the admissibility of right side contraction of formulas c_R (see figure 3.2.4). For the substitution rule, recall Definition 2.1.

Lemma 3.7 (Admissible Rules)**.** *The rules* wk*,* sub *and* c_L*,* c_R *(see figure 3.2.4) are hp-admissible, and* cut *is eliminable in m-G3i.*

Proof. The admissibility of wk, sub, c_L and c_R can be done by a straightforward induction on the height of the derivation. For the elimination of cut see [53, Section 4.1.10]. ☐

$$\text{wk}\frac{\Gamma \Rightarrow \Delta}{\Sigma, \Gamma \Rightarrow \Delta, \Pi} \qquad \text{sub}\frac{\Gamma \Rightarrow \Delta}{\Gamma_{[A/p]} \Rightarrow \Delta_{[A/p]}}$$

$$\text{c}_L\frac{\Gamma, A, A \Rightarrow \Delta}{\Gamma, A \Rightarrow \Delta} \qquad \text{c}_R\frac{\Gamma \Rightarrow \Delta, A, A}{\Gamma \Rightarrow \Delta, A}$$

$$\text{cut}\frac{\Gamma \Rightarrow \Delta, A \qquad A, \Sigma \Rightarrow \Pi}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}$$

Figure 3.2.4: Structural rules and the cut rule for m-G3i

Here again, there are different routes towards soundness and completeness. As discussed before, an effective translation between G3i and m-G3i can be utilised, as well as a cut-elimination argument, from which we can gain syntactic completeness. In this case, we refer to the first completeness proof for the equivalent Beth tableau system, which was done semantically.

Theorem 3.8 (Soundness and Completeness of m-G3i)**.** *The system m-G3i is sound and complete with respect to* IPL*. Formally, m-G3i* $\vdash A$ *iff* IPL $\vdash A$ *for any* $A \in \mathcal{L}$*.*

Proof. Corollary 2.3 and theorem 4.1 of [20]. ☐

3.2.3 Labelled Sequent Calculus for IPL and Intermediate Logics

Extending on the sequent formalism for calculi to strengthen their expressibility has been a very successful approach for structural proof theory in recent decades. The two main approaches include extending the language or the structure of sequents. Enriching the structure can be done by introducing nestings (trees of sequents), or allowing for hypersequents (lists of sequents).[4] Besides being useful for modal logics as they can emulate Kripke structures, nested sequents also have a natural implementation for intuitionistic logic (see [21]).

The other common approach of extending sequent calculus by enriching the language employs what might be called a semantic calculus. Originally, Kanger [26] had the idea of using what he

[4]As an example for a system that incorporates both nestings and hypersequents at once, see [11] which introduces a modular proof system for non-normal modal logic.

called ”spotted” formulas to obtain a cut-free calculus for classical $\mathsf{S5}$ which ultimately evolved into what is now known as the labelled formalism for a variety of logics (see [39, 54]).

The basic idea of labelled systems is to incorporate the semantics of the logic into the syntax of the calculus. This makes normal modal logic and intuitionistic propositional logic a generally easy setup for labelled deductive systems. First, we add a set of countable world labels $\mathbb{WL} = \{x, y, z, ...\}$ (also sometimes referred to as *variables*) as well as some relation symbols (usually called R for classical normal modal logic or $\leq$ for intuitionistic logic). Second, we define a labelled language $\mathcal{L}^{lab}$ over the original logical language $\mathcal{L}$. In the case of IPL, we define *labelled formulas* as $x : A$ and $x \leq y$ for some $A \in \mathcal{L}$ and $x, y \in \mathbb{WL}$. Labelled sequents have the form $\mathcal{R}, \Gamma \Rightarrow \Delta$, where $\mathcal{R}, \Gamma$ and Δ are all multisets of labelled formulas. $\mathcal{R}$ only contains formulas of the form $x \leq y$ (relational formulas), while Γ and Δ contain formulas in the form of $x : A$. We say that a label x *occurs* in a sequent if there is some labelled formula $x : A$ (or either $x \leq y$ or $y \leq x$ for relational formulas) which can be found in the sequent.

This allows one to define a proof system for IPL, naturally corresponding to the semantics of Kripke models. The proof system presented here (see figure 3.2.5) was introduced in [15]. Instead of calling it G3I, as it is called in the paper, we give it the name labG3I to make the usage of labels clear and reduce any confusion with G3i. Note that the rule $\supset$R is a special one by requiring that the variable y cannot occur in the conclusion of the rule. We call this condition that y must occur *fresh* inside the premise. Here, a formula A is derivable in the system, i.e. labG3I $\vdash A$, iff $\Rightarrow x : A$ is derivable for any label x.

$$\text{Ax}\,\frac{}{\mathcal{R}, x \leq y, x : p, \Gamma \Rightarrow \Delta, y : p} \qquad \bot\,\frac{}{\mathcal{R}, x : \bot, \Gamma \Rightarrow \Delta}$$

$$\wedge\text{L}\,\frac{\mathcal{R}, x : A, x : B, \Gamma \Rightarrow \Delta}{\mathcal{R}, x : A \wedge B, \Gamma \Rightarrow \Delta} \qquad \wedge\text{R}\,\frac{\mathcal{R}, \Gamma \Rightarrow \Delta, x : A \qquad \mathcal{R}, \Gamma \Rightarrow \Delta, x : B}{\mathcal{R}, \Gamma \Rightarrow \Delta, x : A \wedge B}$$

$$\vee\text{L}\,\frac{\mathcal{R}, x : A, \Gamma \Rightarrow \Delta \qquad \mathcal{R}, x : B, \Gamma \Rightarrow \Delta}{\mathcal{R}, x : A \vee B, \Gamma \Rightarrow \Delta} \qquad \vee\text{R}\,\frac{\mathcal{R}, \Gamma \Rightarrow \Delta, x : A, x : B}{\mathcal{R}, \Gamma \Rightarrow \Delta, x : A \vee B}$$

$$\supset\text{L}\,\frac{\mathcal{R}, x \leq y, x : A \supset B, \Gamma \Rightarrow \Delta, y : A \qquad \mathcal{R}, x \leq y, x : A \supset B, y : B, \Gamma \Rightarrow \Delta}{\mathcal{R}, x \leq y, x : A \supset B, \Gamma \Rightarrow \Delta}$$

$$\supset\text{R}\,\frac{\mathcal{R}, x \leq y, y : A, \Gamma \Rightarrow \Delta, y : B}{\mathcal{R}, \Gamma \Rightarrow \Delta, x : A \supset B}\,(y \text{ fresh})$$

$$Ref\,\frac{\mathcal{R}, x \leq x, \Gamma \Rightarrow \Delta}{\mathcal{R}, \Gamma \Rightarrow \Delta} \qquad Trans\,\frac{\mathcal{R}, x \leq z, x \leq y, y \leq z, \Gamma \Rightarrow \Delta}{\mathcal{R}, x \leq y, y \leq z, \Gamma \Rightarrow \Delta}$$

Figure 3.2.5: Rules for labG3I

Lemma 3.9. *The general axiom rule*

$$Ax^g \frac{}{\mathcal{R}, x \leq y, \Gamma, x : A \Rightarrow y : A, \Delta}$$

is derivable in labG3I.

Proof. Lemma 2 in [15]. □

A crucial difference between all the previous intuitionistic calculi and labG3I is that all the rules of the system are invertible.

Definition 3.10 (Rule Invertibility)**.** *A (proper) rule, with premises $\mathfrak{S}_1, ..., \mathfrak{S}_n$ $(n \in \{1, 2\})$ and a conclusion $\mathfrak{C}$, is invertible in a system L iff: If the rule*

$$\frac{\mathfrak{C}}{\mathfrak{S}_i}$$

is admissible for any $\mathfrak{S}_i$.

The invertibility of the rules of labG3I, especially of the rule ⊃R, follows immediately from the fact that we do not lose any context (unlike for example in the same rule for m-G3i). Together the admissibility of other structural rules is stated in the following proposition.

Proposition 3.11 (Admissibility in labG3I)**.** *The rules of weakening (*wk*), substitution (*sub_V*), contraction (c_L, c_R, $c^{\leq}$) and* cut *(see figures 3.2.4 and 3.3.2) are admissible in* labG3I*. Also, all rules are invertible.*

Proof. Propositions 2.1 and 2.2 from [15]. □

The invertibility of rules also means that any rule can be applied without creating a wrong branch in which one has to backtrack. It furthermore implies that rules might be permutated in a derivation as long as they do not interfere with other rules that share the same principal formulas. This allows one to obtain a more straightforward proof search algorithm, although the downside to this is that such an algorithm has to incorporate loop checking procedures as we are able to apply certain rules (⊃L, Ref, $Trans$) as often as we want. Thus, when considering an invalid formula, we might end up applying the same rules over and over again without ever needing to stop.

Another strong feature of semantic calculi is that we can introduce extensions of our base logic easily into our system. Naturally, this is interesting for frame definable classes of modal logics, but it also allows us to describe intermediate (superintuitionistic) logics that extend IPL while not being classical. For example, we can easily describe the Gödel–Dummett Logic (GD) which extend IPL by the axiom schema $(A \supset B) \vee (B \supset A)$ and corresponds to the class of *strongly connected* Kripke models, i.e. satisfying the following frame property.

If $x \leq y$ and $x \leq z$ then $y \leq z$ or $z \leq y$.

So, by just adding the following rule to labG3I, we will have obtained a system for GD.

$$\text{GD} \frac{\mathcal{R}, x \leq y, x \leq z, y \leq z, \Gamma \Rightarrow \Delta \qquad \mathcal{R}, x \leq y, x \leq z, z \leq y, \Gamma \Rightarrow \Delta}{\mathcal{R}, x \leq y, x \leq z, \Gamma \Rightarrow \Delta}$$

This also works modularly, so we can emulate adding certain axioms to the logic by adding more rules to the calculus. For more intermediate logics that are definable in this formalism see [15].

The rule of cut is also admissible in labG3I, so we gain the following syntactic completeness result.

Theorem 3.12 (Soundness and Completeness of labG3I)**.** *The system labG3I is sound and complete wrt.* IPL*, meaning that for any formula* $A \in \mathcal{L}$*:* labG3I $\vdash A$ *iff* IPL $\vdash A$*.*

Proof. Soundness can be established by theorem 3.4 in [40].

For completeness, consider first that we can derive all schemata of the Hilbert axiomatisation for IPL in labG3I, utilising the admissibilty of Ax^g. Here is one example case:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{x \le y, y \le z, y : A, z : B \Rightarrow z : A}\,\text{Ax}^g \qquad \dfrac{}{x \le y, y \le z, y : A, z : B \Rightarrow z : B}\,\text{Ax}^g}{x \le y, y \le z, y : A, z : B \Rightarrow z : A \wedge B}\,\wedge\text{R}}{x \le y, y : A \Rightarrow y : B \supset (A \wedge B)}\,\supset\text{R}}{\Rightarrow x : A \supset (B \supset (A \wedge B))}\,\supset\text{R}$$

Now, via the admissibility of cut (proposition 3.11), we immediately gain completeness wrt. the Hilbert axiomatisation. □

A semantic calculus allows us to have an easy counter model construction, which also allows us to gain completeness semantically, as discussed in the previous section. Such a semantic completeness result has been formulated by Negri in [40] for some intermediate logics (extensions of IPL with geometric implications). The result establishes completeness wrt. sequents, as simply opposed to formulas, as any labelled sequent can be interpreted in a model. Another such completeness result was obtained in [38] for classical modal logics inside the S5 cube. Consider the following example for a failed proof in labG3I.

Example 3.13 (Countermodel construction from a failed proof of $\neg p \vee \neg\neg p$)**.** We use the following rules for negation that are admissible in labG3I.

$$\neg\text{R}\,\dfrac{\mathcal{R}, x \le y, y : A, \Gamma \Rightarrow \Delta}{\mathcal{R}, \Gamma \Rightarrow \Delta, x : \neg A}\,(y \text{ fresh}) \qquad\qquad \neg\text{L}\,\dfrac{\mathcal{R}, x \le y, x : \neg A, \Gamma \Rightarrow \Delta, y : A}{\mathcal{R}, x \le y, x : \neg A, \Gamma \Rightarrow \Delta}$$

Equivalently, one could apply the respective $\supset$ rule and possibly $\perp$ (note that any formula $x : \perp$ in the succedent can be omitted).

One failed proof of this formula in labG3I can then be displayed as follows.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdots}{x \le y, x \le z, x \le x, y \le y, z \le z, y : p, z : \neg p \Rightarrow z : p}\,Ref}{x \le y, x \le z, x \le x, z \le z, y : p, z : \neg p \Rightarrow z : p}\,Ref}{x \le y, x \le z, z \le z, y : p, z : \neg p \Rightarrow z : p}\,\neg\text{L}}{x \le y, x \le z, z \le z, y : p, z : \neg p \Rightarrow}\,Ref}{x \le y, x \le z, y : p, z : \neg p \Rightarrow}\,\neg\text{R}}{x \le y, y : p \Rightarrow x : \neg\neg p}\,\neg\text{R}}{\Rightarrow x : \neg p, x : \neg\neg p}\,\vee\text{R}}{\Rightarrow x : \neg p \vee \neg\neg p}$$

In the top sequent are no more rule applications available that would give us new formulas. Therefore, continuing by using the algorithm presented in [40], we would get an infinite branch, so we can read out the finite counter model from the topmost sequent: $\mathcal{M} = \langle W, \leq, V \rangle$ with $W = \{x, y, z\}$, $\leq = \{(x, y), (x, z)\}^{\star}$ and $V(p) = \{y\}$, where $\star$ denotes the reflexive-transitive closure.[5] It should now be fairly easy to see that $\mathcal{M}, x \nVdash \neg p \vee \neg\neg p$.

One thing in which labG3I works differently than usual sequent systems is the way the axiom sequent is formulated. For example, in classical modal logic an initial sequent always contains the same labelled atomic formula with the same label, as opposed to here, where we include the $\leq$ relation, such that the rule corresponds to the monotonicity condition in Kripke models (or the translations of atoms into boxed atoms in the Gödel translation for S4). But what about an explicit rule that allows us to move the truth of propositions into $\leq$-related states? As it turns out, this rule which we write down as

$$\text{mon}_L \frac{\mathcal{R}, x \leq y, x : p, y : p, \Gamma \Rightarrow \Delta}{\mathcal{R}, x \leq y, x : p, \Gamma \Rightarrow \Delta}$$

is indeed admissible for the system. The issue with including it explicitly, though, is that it would cause problems for establishing cut-elimination, just like contraction which we have to admit first before eliminating cuts. A similar rule

$$\text{mon}_R \frac{\mathcal{R}, x \leq y, \Gamma \Rightarrow \Delta, y : p, x : p}{\mathcal{R}, \Gamma \Rightarrow \Delta, y : p}$$

can be seen as the dual of mon_L and is also admissible (see lemma 3.15 where labG3I can be seen as a modal-free case of labIK).

3.3 Intuitionistic Modal Logic

Proof systems for logics that are both intuitionistic and have modalities exist for as long as such logics have been considered. The first occurrence of both was in 1948 in a paper by Fitch [19], where he introduced an axiomatisation for a first-order modal logic based on the classical modal logic T. Later, Prior introduced in [48] what he called MIPQ (also named MIPC), which is an intuitionistic version of S5 and is the first logic falling under what we would now call "proper" intuitionistic modal logic. A decision procedure in the style of Gentzen was later developed by Minc in a Russian article (see [51] for reference). Further systems for different logics include different intuitionistic versions of S4 and S5 (see [41, 47]).

Many of these previous approaches, though, did not really take off or have much influence. This is mainly due to their disagreement at the time on what is "the" (main) intuitionistic modal logic to consider. Another apparent issue is that finding a Gentzen-style system for IK, which we showed is what comes closest to a basic intuitionistic modal logic, is at least non-trivial and maybe even impossible, as far as we know.[6] This makes Simpson's approach of using a semantic calculus in the style of Kanger (i.e. using labels to incorporate modalities) a milestone for proof theory of

[5] The fact that there is no labelled formula $x : p$ occurring in the open leaf tells us that we have a free choice on the evaluation for x.

[6] Notice, that Simpson's calculus in [50] might not be considered *Gentzen-style* as it relies on additional relational formulas. This also holds for the equivalent natural deduction system that is found in the same thesis.

intuitionistic modal logic. Since then, a multitude of systems have been studied and introduced, most of them based on some intuitionistic system from the previous section. We might, for instance, combine the approaches we gathered for intuitionistic and modal calculi to gain systems that can handle both things, modalities and being intuitionistic, at the same time. As an example, Simpson's calculus, from [50], is essentially obtained by taking G3i and adding relational labels. Similarly, we will consider doing the same to labG3I, giving us what we call a *fully labelled* calculus. We also include another recently introduced system in the style of Maehara's calculus extended with nestings.

3.3.1 Labelled Calculi

Similarly to how we used semantic terms for a calculus of IPL, we can incorporate the full semantics of IK, namely bi-relational Kripke models, to obtain an analytic proof system for intuitionistic modal logic. Such a system has been recently introduced by Marin, Morales, and Straßburger in [36], where the authors also show completeness for IK with extensions of one-sided intuitionistic Scott-Lemmon axioms. These axioms form a proper superset of HSL while still being contained in the set of Scott-Lemmon axioms.

For the language of this fully labelled (or *bi-labelled*) system, labIK,[7] we can simply extend the language of labG3I with an additional relational symbol R. So, together with a set of world labels $\mathbb{WL}$ and the logical language $\mathcal{L}^{\Box\Diamond}$, we have labelled formulas $x : A$, xRy, and $x \leq y$ with $x, y \in \mathbb{WL}$ and $A \in \mathcal{L}^{\Box\Diamond}$. The rules are given in figure 3.3.1.

Notice that, besides adding modal rules that correspond to the respective truth conditions, we also have to bake in the confluence conditions of the Kripke models with the rules F_1 and F_2.

Similar to formula substitution, we define a *variable substitution* $\Gamma_{[x/y]}$ for a multiset of labelled formulas Γ by replacing the world label y in any formula in which it occurs by the world label x.

Now we are ready to state the major structural results of labIK.

Proposition 3.14. *The general axiom schema*

$$\frac{}{\mathcal{R}, x \leq y, x : A, \Gamma \Rightarrow \Delta, y : A}\ Ax^g$$

is admissible in labIK.

Proof. See proposition 3.2 of [36]. □

Lemma 3.15 (Monotonicity)**.** *The rules* mon_L*,* mon_R *and* mon_L^g *(see figure 3.3.2) are admissible in* labIK*.*

Proof. Lemma 6.3 and proposition 3.1 in [36]. □

Notice that the general monotonicity rule also implies contraction for labelled formulas $x : A$, via the use of Ref.

Lemma 3.16 (Weakening, Substitution and Contraction)**.** *The rules of weakening* (wk)*, label substitution* (sub_V) *and contraction for relational formulas (*c^R *and* $c^{\leq}$*) (see 3.3.2) are hp-admissible in labIK.*

[7] In [36] the system is called $\text{labIK}_{\leq}$.

Initial Sequents

$$\text{Ax}\,\frac{}{\mathcal{R}, x \leq y, x : p, \Gamma \Rightarrow \Delta, y : p} \qquad \bot\,\frac{}{\mathcal{R}, x : \bot, \Gamma \Rightarrow \Delta}$$

Rules for Connectives

$$\wedge\text{L}\,\frac{\mathcal{R}, x : A, x : B, \Gamma \Rightarrow \Delta}{\mathcal{R}, x : A \wedge B, \Gamma \Rightarrow \Delta} \qquad \wedge\text{R}\,\frac{\mathcal{R}, \Gamma \Rightarrow \Delta, x : A \qquad \mathcal{R}, \Gamma \Rightarrow \Delta, x : B}{\mathcal{R}, \Gamma \Rightarrow \Delta, x : A \wedge B}$$

$$\vee\text{L}\,\frac{\mathcal{R}, x : A, \Gamma \Rightarrow \Delta \qquad \mathcal{R}, x : B, \Gamma \Rightarrow \Delta}{\mathcal{R}, x : A \vee B, \Gamma \Rightarrow \Delta} \qquad \vee\text{R}\,\frac{\mathcal{R}, \Gamma \Rightarrow \Delta, x : A, x : B}{\mathcal{R}, \Gamma \Rightarrow \Delta, x : A \vee B}$$

$$\supset\text{L}\,\frac{\mathcal{R}, x \leq y, x : A \supset B, \Gamma \Rightarrow \Delta, y : A \qquad \mathcal{R}, x \leq y, x : A \supset B, y : B, \Gamma \Rightarrow \Delta}{\mathcal{R}, x \leq y, x : A \supset B, \Gamma \Rightarrow \Delta}$$

$$\supset\text{R}\,\frac{\mathcal{R}, x \leq y, y : A, \Gamma \Rightarrow \Delta, y : B}{\mathcal{R}, \Gamma \Rightarrow \Delta, x : A \supset B}(y \text{ fresh})$$

$\leq$ is a Pre-Order

$$Ref\,\frac{\mathcal{R}, x \leq x, \Gamma \Rightarrow \Delta}{\mathcal{R}, \Gamma \Rightarrow \Delta} \qquad Trans\,\frac{\mathcal{R}, x \leq z, x \leq y, y \leq z, \Gamma \Rightarrow \Delta}{\mathcal{R}, x \leq y, y \leq z, \Gamma \Rightarrow \Delta}$$

Modal Rules

$$\Box\text{L}\,\frac{\mathcal{R}, x \leq y, yRz, x : \Box A, z : A, \Gamma \Rightarrow \Delta}{\mathcal{R}, x \leq y, yRz, x : \Box A, \Gamma \Rightarrow \Delta}$$

$$\Box\text{R}\,\frac{\mathcal{R}, x \leq y, yRz, \Gamma \Rightarrow \Delta, z : A}{\mathcal{R}, \Gamma \Rightarrow \Delta, x : \Box A}(y, z \text{ fresh})$$

$$\Diamond\text{L}\,\frac{\mathcal{R}, xRy, y : A, \Gamma \Rightarrow \Delta}{\mathcal{R}, x : \Diamond A, \Gamma \Rightarrow \Delta}(y \text{ fresh}) \qquad \Diamond\text{R}\,\frac{\mathcal{R}, xRy, \Gamma \Rightarrow \Delta, x : \Diamond A, y : A}{\mathcal{R}, xRy, \Gamma \Rightarrow \Delta, x : \Diamond A}$$

Confluence Rules

$$\text{F}_1\,\frac{\mathcal{R}, xRy, y \leq z, x \leq u, uRz, \Gamma \Rightarrow \Delta}{\mathcal{R}, xRy, y \leq z, \Gamma \Rightarrow \Delta}(u \text{ fresh})$$

$$\text{F}_2\,\frac{\mathcal{R}, xRy, x \leq z, y \leq u, zRu, \Gamma \Rightarrow \Delta}{\mathcal{R}, xRy, x \leq z, \Gamma \Rightarrow \Delta}(u \text{ fresh})$$

Figure 3.3.1: Rules for labIK

$$\text{wk}\frac{\mathcal{R},\Gamma\Rightarrow\Delta}{\mathcal{R},\mathcal{R}',\Sigma,\Gamma\Rightarrow\Delta,\Pi} \qquad \text{sub}_V\frac{\mathcal{R},\Gamma\Rightarrow\Delta}{\mathcal{R}_{[x/y]},\Gamma_{[x/y]}\Rightarrow\Delta_{[x/y]}}\dagger$$

$$\text{mon}_L\frac{\mathcal{R},x\le y,x:p,y:p,\Gamma\Rightarrow\Delta}{\mathcal{R},x\le y,x:p,\Gamma\Rightarrow\Delta} \qquad \text{mon}_R\frac{\mathcal{R},x\le y,\Gamma\Rightarrow\Delta,y:p,x:p}{\mathcal{R},x\le y,\Gamma\Rightarrow\Delta,y:p}$$

$$\text{mon}_L^g\frac{\mathcal{R},x\le y,x:A,y:A,\Gamma\Rightarrow\Delta}{\mathcal{R},x\le y,x:A,\Gamma\Rightarrow\Delta}$$

$$\text{c}^R\frac{\mathcal{R},xRy,xRy,\Gamma\Rightarrow\Delta}{\mathcal{R},xRy,\Gamma\Rightarrow\Delta} \qquad \text{c}^{\le}\frac{\mathcal{R},x\le y,x\le y,\Gamma\Rightarrow\Delta}{\mathcal{R},x\le y,\Gamma\Rightarrow\Delta}$$

($\dagger$: x is not in $\mathcal{R},\Gamma\Rightarrow\Delta$)

Figure 3.3.2: Structural rules for labIK

Proof. For the admissibility of wk see lemma 6.2 in [36]. Just like weakening, contraction also follows immediately from the fact that all context formula are being upwards preserved.

For sub_V, we do a straightforward proof by induction on the height of the derivation. Consider only rules that introduce new variables, as all other ones become trivial. We consider here only the cases of $\Box$R and F_2, as all others work in the same way or can at least be obtained with the steps presented.

$\Box$R: Assume $\mathcal{R},\Gamma\Rightarrow\Delta,y:\Box A$ is provable in at least n steps with $y:\Box A$ as the principal formula going upwards in the proof. Then, $\mathcal{R},yRu,u\le w,\Gamma\Rightarrow\Delta,w:A$ is provable in $n-1$ steps. By induction hypothesis, we know that the label x does not occur in $\mathcal{R},yRu,u\le w,\Gamma\Rightarrow\Delta,w:A$ and we can replace y with x in the whole sequent and leave it provable on the same number of steps. Thus, $\mathcal{R}_{[x/y]},xRu,u\le w,\Gamma_{[x/y]}\Rightarrow\Delta_{[x/y]},w:A$ is provable in $n-1$ steps. Applying $\Box$R here, we get $\mathcal{R}_{[x/y]},\Gamma_{[x/y]}\Rightarrow\Delta_{[x/y]},x:\Box A$ after n steps.

F_2: Assume $\mathcal{R},xRy,y\le z,\Gamma\Rightarrow\Delta$ is provable in n steps and w does not occur in any other formula of the sequent. Further assume that xRy and $y\le z$ are principal. Then, $\mathcal{R},xRy,y\le z,y\le u,zRu,\Gamma\Rightarrow\Delta$ is provable after $n-1$ steps. By induction hypothesis and again applying F_2, the following are all provable after n steps:

$\mathcal{R}_{[w/x]},wRy,y\le z,\Gamma_{[w/x]}\Rightarrow\Delta_{[w/x]}$
$\mathcal{R}_{[w/y]},xRw,w\le z,\Gamma_{[w/y]}\Rightarrow\Delta_{[w/y]}$
$\mathcal{R}_{[w/z]},xRy,y\le w,\Gamma_{[w/z]}\Rightarrow\Delta_{[w/z]}$

This covers all cases of substitution which cannot use the induction hypothesis trivially. $\Box$

Besides these more common structural properties, the system of labIK has the following strong property.

Lemma 3.17 (Invertibility)**.** *All rules of* labIK *are invertible.*

Proof. Most rules can be shown to be invertible by simply using weakening ($\supset$L, $\Box$L, $\Diamond$R and the relational rules Ref, $Trans$, F_1 and F_2). For all other rules, it simply follows from the fact that the context of sequents is always preserved and all relational formulas are unique, meaning that some new formula $x \leq y$ or xRy gets only introduced if at least y is fresh. This makes all relational formulas distinguishable from one another and therefore connects them to some unique labelled formulas. □

This lemma essentially allows us to apply rules in any order we want; unlike in m-G3i, where one has to be careful with the order rules are being applied in (see also examples 4.13 and 4.29).

Extending the logic IK with the axioms from the same set of HSLs and axiom D, can be easily achieved for labIK by adding a specific rule for every axiom (see figure 3.3.3). These rules correspond directly to the frame conditions we discussed in chapter 2. Also, just like with the frame conditions, the general rule reduces itself when one of the indices is zero by simply leaving out xR^0u and replacing u with x.

$$\mathrm{d}\frac{\mathcal{R}, xRy, \Gamma \Rightarrow \Delta}{\mathcal{R}, \Gamma \Rightarrow \Delta}(y \text{ fresh}) \qquad\qquad S_{n,m}\frac{\mathcal{R}, xR^ny, xR^mz, yRz, \Gamma \Rightarrow \Delta}{\mathcal{R}, xR^ny, xR^mz, \Gamma \Rightarrow \Delta}$$

Figure 3.3.3: Extension rules for labIK($\mathcal{A}$)

Definition 3.18. *The rules of* labIK($\mathcal{A}$) *are the rules of* labIK *together with the rule* d *iff* $D \in \mathcal{A}$ *and* $S_{n,m}$ *iff* $\phi(n,m) \in \mathcal{A}$ *for any* $n, m \in \mathbb{N}$.

All the structural results we have obtained so far also hold for the general system labIK($\mathcal{A}$) as proving them also for the additional rules becomes trivial since they only have relational formulas as principal formulas.

Theorem 3.19 (Soundness and Completeness)**.** *The system* labIK($\mathcal{A}$) *is sound and complete wrt. to* $\mathsf{IK} \oplus \mathcal{A}$*, i.e.* labIK($\mathcal{A}$) $\vdash B$ *iff* $\mathsf{IK} \oplus \mathcal{A} \vdash B$ *for any formula* $B \in \mathcal{L}^{\Box\Diamond}$.

Proof. Special case of theorem 7.2 of [36]. □

As previously mentioned, it is possible to show completeness via the semantic route and counter model construction. Though, especially with the more complex bi-relational structures, it is harder to formulate a terminating procedure which takes care of loop checking (all rules are invertible). Such a completeness result was recently found by Girlando et al. in [24] for the logic IS4, which is also the first decidability result for this logic, as well as for IK in [23].

3.3.2 Nested Calculi

Another successful approach to proof-theoretically emulate modalities, which some might think as less controversial due to the omission of semantic terms, is the use of nested sequents, the basic idea being that we can write sequents inside some nesting $[\Gamma \Rightarrow \Delta]$, thereby essentially "modalizing" or "boxing" whole sequents. This builds on the idea that we want to interpret sequents as formulas; thus, sequents inside nestings can be classically interpreted as boxed formulas, like $fm([\Gamma \Rightarrow \Delta]) = \Box(\bigwedge \Gamma \rightarrow \bigvee \Delta)$. We did not have this for labelled systems, as we cannot interpret relations and labels into modal formulas. However, the formula interpretation does not work or at least works differently in non-classical systems. Here, we consider so called *simple nested* sequents, meaning that we only have a single type of nesting. By using, for instance, two types of nestings we could construct a notational variant of labIK, where the two nestings correspond to the two relations R and $\leq$ (see also [43]).

Just like the labelled formalism, nested sequent calculus developed from different ideas that preceded it. This includes for example the hypersequent calculus introduced by Avron [1], the display calculus due to Belnap [4] and the idea of making deep inferences as done already by Schütte [49].[8] This is why nested sequent systems also get referred to as *deep inference* systems, as they can be thought of as a version of display calculus in which one can apply rules inside the nesting.[9] Nested sequents were first introduced in their current form by Brünnler in [8] for classical modal logic in the S5 cube. Nested sequents were also independently discovered by Bull [9], Kashima [27] and Poggiolesi [46] under different formalisms. Here, we will consider a single- and multi-succedent variant of Brünnler's calculus, essentially weaving nestings into the systems G3i and m-G3i, which were first presented in [52] and [31], respectively.

To have better readability, Brünnler uses one-sided sequents, in which all formulas can only occur inside the succedent of the sequent. But, as discussed before, this relies on De-Morgan laws which we do not have in intuitionism. Instead, systems based on intuitionistic logic often use polarisations of formulas, $A^\bullet$ and A°, which indicates whether the formula is inside the antecedent ($\bullet$) or succedent ($\circ$). Formulas polarised by $\circ$ are also called *output* formulas, while $\bullet$-formulas are called *input* formulas. Polarised sequents are solely a notational variant of "proper" two-sided sequents and serve solely the purpose of readability. Thus, polarised nested sequents, henceforth called *nested sequents*, are constructed by the following grammar (see [31]).

$$\Gamma, \Delta ::= \emptyset \mid \Gamma, A^\bullet \mid \Gamma, A^\circ \mid \Gamma, [\Delta]$$

Such sequents can be used for classical or intuitionistic multi-succedent calculi while, in most intuitionistic calculi (e.g. [52]), single-output sequents with only one $\circ$-formula are needed. Observe that all nested sequents will be of the form $\Gamma, [\Delta_1], ..., [\Delta_n]$ where Γ is nesting free and Δ_i is always some sequent, possibly with nestings.

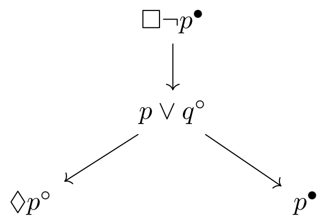
Definition 3.20 (Sequent Tree)**.** *For a nested sequent* Γ *we define its tree* $tr(\Gamma)$ *inductively by*

1. *the multiset of all unnested formulas of* Γ *form the root of* $tr(\Gamma)$ *and*
2. *for any* $[\Delta]$ *occurring in* Γ*,* $tr(\Delta)$ *occurs in the tree of* Γ *such that the root of* $tr(\Delta)$ *is a child node of* Γ*.*

Example 3.21. The sequent $\Box\neg p^\bullet, [[\Diamond p^\circ], [p^\bullet], p \vee q^\circ]$ is a polarised nested sequent which could also be written as $\Box\neg p \Rightarrow [\Rightarrow p \vee q, [\Rightarrow \Diamond p], [p \Rightarrow]]$. It gets classically interpreted into the following formula $fm(\Box\neg p^\bullet, [[\Diamond p^\circ], [p^\bullet], p \vee q^\circ]) = \Box\neg p \to \Box((p \vee q) \vee \Box\neg p \vee \Box\Diamond p)$. The sequent tree looks

[8] For a survey of structural proof systems, like hypersequent or display calculi, see [55].
[9] A simple example of a (shallow) display calculus for modal tense logic can be found in [27].

as follows.



Due to nested sequents essentially being trees of sequents, this nested formalism is equivalent to the so-called *tree-hypersequent* formalism, where sequents are trees of basic sequents. It furthermore has been shown for classical modal logic that nested sequent proofs translate constructively into labelled ones, as long as the relational labels form a tree-like structure (see [25]).

To define a deep nested proof system, we have to introduce the concept of context and holes for nested sequents to be able to apply rules anywhere in the sequent.

Definition 3.22 (Sequent Context)**.** *A sequent with a hole (context sequent), written as* $\Gamma\{\}$*, is a nested sequent containing a hole* $\{\}$ *as one of its elements and is inductively defined on the depth of the hole* $depth(\Gamma\{\})$

- $\Gamma, \{\}$ *is a context sequent with* $depth(\Gamma\{\}) = 0$*.*

- *If* $\Pi\{\}$ *is a context sequent with* $depth(\Pi\{\}) = m$ *then* $\Sigma, [\Pi\{\}]$ *is a context sequent with* $depth(\Gamma\{\}) = m + 1$*.*

We write $\Gamma\{\Delta\}$ *when replacing the hole in* $\Gamma\{\}$ *by* Δ*, where* Δ *is itself a (possibly nested) sequent. If the hole is replaced by an empty sequent, we also write* $\Gamma\{\}$ *instead of* $\Gamma\{\emptyset\}$*.*

Similar to two-sided sequents and their rules, the context is defined by the non-active formulas in rule applications; that is usually the part outside of the hole. Besides that, *context* in a nested sequent $\Gamma\{\Delta\}$ can also refer to the hole-input, in this case Δ. We also allow for sequents with several contexts, which we can define by reiterating the process of defining a context in a sequent with an already existing context. When talking about the depth in a multi-context sequent, we will clarify the meaning by explicitly stating which one has an empty input. For example, $depth(\Gamma\{\}\{\emptyset\})$ asks for the nesting depth of the first hole of $\Gamma\{\}\{\}$.

Because we are extending the multi-conclusion calculus m-G3i in which we sometimes lose succedent formulas in rule applications, we define the *output pruning* for context sequents.

Definition 3.23 (Output Pruning)**.** *For a sequent with context* $\Gamma\{\}$*, we obtain the output pruning* $\Gamma^{\downarrow}\{\}$ *by removing all output formulas in* $\Gamma\{\}$*.*

The inference rules for m-NIK (given in figure 3.3.4 and based on [31], where it is called NIKm) are essentially the same as for m-G3i for the non-modal cases. Notice that, besides the $\supset^\circ$ rule, we have a second rule $\Box^\circ$ in which we get rid of output formulas. We also consider here a *contraction variant* from [31], which means that some rules repeat their principal formulas. This corresponds well to the labelled system labIK from the previous section. This also means that we can have contraction (left and right) implicitly as admissible rules.

The structural properties of m-NIK are not as well explored as for other already more established systems. Still, a few results are present in the currently only paper where it was introduced (see [31]).

Proposition 3.24. *The rule* wk *is hp-admissible and* Ax^g *is derivable in* m-NIK.

Initial Sequents

$$\text{Ax}\,\frac{}{\Gamma\{p^\bullet, p^\circ\}} \qquad \bot\,\frac{}{\Gamma\{\bot^\bullet\}}$$

Rules for Connectives

$$\wedge^\bullet\frac{\Gamma\{A^\bullet, B^\bullet\}}{\Gamma\{A \wedge B^\bullet\}} \qquad \wedge^\circ\frac{\Gamma\{A^\circ\} \qquad \Gamma\{B^\circ\}}{\Gamma\{A \wedge B^\circ\}} \qquad \vee^\bullet\frac{\Gamma\{A^\bullet\} \qquad \Gamma\{B^\bullet\}}{\Gamma\{A \vee B^\bullet\}} \qquad \vee^\circ\frac{\Gamma\{A^\circ, B^\circ\}}{\Gamma\{A \vee B^\circ\}}$$

$$\supset^\bullet\frac{\Gamma\{A \supset B^\bullet, A^\circ\} \qquad \Gamma\{A \supset B^\bullet, B^\bullet\}}{\Gamma\{A \supset B^\bullet\}} \qquad \supset^\circ\frac{\Gamma^\downarrow\{A^\bullet, B^\circ\}}{\Gamma\{A \supset B^\circ\}}$$

Modal Rules

$$\Box^\bullet\frac{\Gamma\{\Box A^\bullet, [\Delta, A^\bullet]\}}{\Gamma\{\Box A^\bullet, [\Delta]\}} \qquad \Box^\circ\frac{\Gamma^\downarrow\{[A^\circ]\}}{\Gamma\{\Box A^\circ\}} \qquad \Diamond^\bullet\frac{\Gamma\{[A^\bullet]\}}{\Gamma\{\Diamond A^\bullet\}} \qquad \Diamond^\circ\frac{\Gamma\{\Diamond A^\circ, [\Delta, A^\circ]\}}{\Gamma\{\Diamond A^\circ, [\Delta]\}}$$

Figure 3.3.4: Rules for m-NIK

$$\text{Ax}^g\,\frac{}{\Gamma\{A^\bullet, A^\circ\}} \qquad \text{wk}\,\frac{\Gamma\{\emptyset\}}{\Gamma\{\Delta\}} \qquad \text{cut}\,\frac{\Gamma\{A^\circ\} \qquad \Gamma\{A^\bullet\}}{\Gamma\{\emptyset\}}$$

Figure 3.3.5: General axiom, weakening and cut-rule for m-NIK

Proof. Lemma 3.11 in [31]. Derivability of Ax^g can be shown by induction on A. Consider $\Box A$ and $A \supset B$ as key cases:

$$\dfrac{\dfrac{\dfrac{\vdots}{\Gamma^\downarrow\{[A^\bullet, A^\circ]\}}\,\text{I.H.}}{\Gamma^\downarrow\{\Box A^\bullet, [A^\circ]\}}\,\Box^\bullet}{\Gamma\{\Box A^\bullet, \Box A^\circ\}}\,\Box^\circ \qquad \dfrac{\dfrac{\dfrac{\vdots}{\Gamma^\downarrow\{A \supset B^\bullet, A^\circ, A^\bullet, B^\circ\}}\,\text{I.H.} \qquad \dfrac{\vdots}{\Gamma^\downarrow\{A \supset B^\bullet, B^\bullet, A^\bullet, B^\circ\}}\,\text{I.H.}}{\Gamma^\downarrow\{A \supset B^\bullet, A^\bullet, B^\circ\}\}}\,\supset^\bullet}{\Gamma\{A \supset B^\bullet, A \supset B^\circ\}}\,\supset^\circ$$

$\Box$

Extending nested systems is much less straightforward than for semantic calculi due to our inability of accessing the nesting tree directly. This especially applies to general extensions with sets of axioms like Scott-Lemmon. But what has been studied extensively for nested systems are extensions of the $\mathsf{S5}$ cube (see figure 2.2.2).

Definition 3.25. *For a set of formulas* $\mathcal{A} \subseteq \{D, B, T, 4, 5\}$ *we call* m-NIK($\mathcal{A}$) *the system* m-NIK *extended with the following rules (see figure 3.3.6).*

$d^{[\,]}$ *if* $D \in \mathcal{A}$,

t° and $t^\bullet$ if $T \in \mathcal{A}$,
b° and $b^\bullet$ if $B \in \mathcal{A}$,
4° and $4^\bullet$ if $4 \in \mathcal{A}$, and
5° and $5^\bullet$ if $5 \in \mathcal{A}$.

$$\mathrm{d}^{[\,]}\frac{\Gamma\{[\,]\}}{\Gamma\{\emptyset\}} \qquad \mathrm{t}^\circ\frac{\Gamma\{\Diamond A^\circ, A^\circ\}}{\Gamma\{\Diamond A^\circ\}} \quad \mathrm{b}^\circ\frac{\Gamma\{[\Delta, \Diamond A^\circ], A^\circ\}}{\Gamma\{[\Delta, \Diamond A^\circ]\}} \quad 4^\circ\frac{\Gamma\{[\Delta, \Diamond A^\circ], \Diamond A^\circ\}}{\Gamma\{[\Delta], \Diamond A^\circ\}} \quad 5^\circ\frac{\Gamma\{\Diamond A^\circ\}\{\Diamond A^\circ\}}{\Gamma\{\Diamond A^\circ\}\{\emptyset\}}\dagger$$

$$\mathrm{t}^\bullet\frac{\Gamma\{\Box A^\bullet, A^\bullet\}}{\Gamma\{\Box A^\bullet\}} \quad \mathrm{b}^\bullet\frac{\Gamma\{[\Delta, \Box A^\bullet], A^\bullet\}}{\Gamma\{[\Delta, \Box A^\bullet]\}} \quad 4^\bullet\frac{\Gamma\{[\Delta, \Box A^\bullet], \Box A^\bullet\}}{\Gamma\{[\Delta], \Box A^\bullet\}} \quad 5^\bullet\frac{\Gamma\{\Box A^\bullet\}\{\Box A^\bullet\}}{\Gamma\{\Box A^\bullet\}\{\emptyset\}}\dagger$$

$$(\dagger : \mathit{depth}(\Gamma\{\}\{\emptyset\}) > 0)$$

Figure 3.3.6: Extension rules for m-NIK

Unlike most axioms, extending the logic with D can be done with a single rule, instead of having a rule for each side of the sequent. This rule, it turns out, is even necessary if we want our system to have contraction on the right as an admissible rule, as noted in [35]. Still, we can derive a corresponding pair of rules that match the structure of the other ones. In the case of d°, it looks like the following, while the case for $\mathrm{d}^\bullet$ is similar.

$$\mathrm{d}^\circ\frac{\Gamma\{\Diamond A^\circ, [A^\circ]\}}{\Gamma\{\Diamond A^\circ\}} \quad \rightsquigarrow \quad \dfrac{\dfrac{\Gamma\{\Diamond A^\circ, [A^\circ]\}}{\Gamma\{\Diamond A^\circ, [\,]\}}\,\Diamond^\circ}{\Gamma\{\Diamond A^\circ\}}\,\mathrm{d}^{[\,]}$$

Brünnler noted in [8] that, though such a nested system is mostly extendable in a modular way, it does not work out fully. He showed that his classical nested calculus extended with the rules for T and 5 cannot prove axiom 4, though it follows from $\mathsf{S5}$. To circumnavigate this issue, he showed that it is enough to consider a *45-closed* (also *45-complete*) rule set.

Definition 3.26. *Let $\mathcal{A} \subseteq \{D, B, T, 4, 5\}$. $\mathcal{A}$ is called 45-closed if the following conditions are fulfilled:*

1. *if $\mathsf{IK} \oplus \mathcal{A} \vdash 4$ then $4 \in \mathcal{A}$*
2. *if $\mathsf{IK} \oplus \mathcal{A} \vdash 5$ then $5 \in \mathcal{A}$.*

This allows us to state the completeness theorem for the nested multi succedent calculus.

Theorem 3.27. *For a 45-closed set $\mathcal{A} \subseteq \{D, B, T, 4, 5\}$ the calculus* m-NIK($\mathcal{A}$) *is sound and complete wrt. the intuitionistic modal logic $\mathsf{IK} \oplus \mathcal{A}$.*

Proof. Theorem 5.3 in [31] together with how 45-closure is defined. □

Remark 3.28. The completeness part of the referenced theorem 5.3 ([31]) relies on a possibly infinite counter model construction. It furthermore establishes soundness and completeness for the provability of sequents and their interpretation in a model. Simple completeness wrt. formulas can also be obtained via a more syntactic route.

Corollary 3.29 (Cut admissibility)**.** *For some 45-closed* $\mathcal{A} \subseteq \{D, B, T, 4, 5\}$ *the rule* cut *is admissible in* m-NIK($\mathcal{A}$)*.*

Proof. See corollary 5.22 in [31]. □

Chapter 4

Proof Translations

In the following sections, we will introduce the general concept of effective translations for derivation trees. Furthermore, we will discuss already established results for translations in modal logic; these are often connecting labelled and nested formalisms. This includes the currently only existing connection between calculi for intuitionistic modal logic. Finally, we will establish a new translation between the multi-succedent nested system m-NIK and the bi-labelled calculus labIK, which we introduced in the previous chapter.

4.1 Translations in General

When considering different calculi for the same class of logics, a natural question to ask is how these systems relate to one another. We may, for example, ask which structural properties they have or how complex their derivations are. One way to explicitly establish a relation is via a constructive (or effective) proof translation.

If we have two different systems for the same logic, we naturally could transform a derivation of one into a proof of the other system by simply doing proof search, but this is a very non-constructive procedure that does not say anything about the structural relationship. We are therefore interested in algorithmic transformations of derivations for which we can explicitly state a general procedure.

In the best case, these translations also allow us to keep structural properties and see, for example, the limits of modularity for different calculi. A classic example for this is the relation between labelled and nested systems for classical modal logic, which have been shown to be notational variants of each other by Goré and Ramanayake in [25]. In their paper, Goré and Ramanayake specifically limit themselves to tree-like labelled structures which correspond to the tree-like structure one obtains when reading out nested sequents. The paper mainly considers systems *induced* by the translation. This means that they considered different systems for the same logic and simply translate the respective rules to gain rules for the other formalism. They specifically compared two systems for the provability logic GL, namely Negri's labelled G3GL (see [39]) and Poggiolesi's tree-hypersequent calculus CSGL (see [46]). These were found to be almost equivalent, except that G3GL does not necessarily rely on a tree structure; G3GL also relies on relational rules that create new structure, while CSGL uses propagation rules which can move formulas along the structure.

The first result on modular translations between calculi for intuitonistic modal logic was obtained in [32]. There, Lyon relates the Simpson calculus to the nested single succedent calculus for the class of logics covered by $\mathsf{IK} \oplus \mathcal{A}$. Essentially, Simpson's calculus $\mathrm{L}_{\Box\Diamond}(\mathcal{A})$ is the $\leq$-free version of our labIK($\mathcal{A}$) together with restricting sequents to a single succedent formula. The nested system, referred to as NIK($\mathcal{A}$), is obtained by restricting m-NIK($\mathcal{A}$) to single succedent sequents. For a proper definition of these calculi, see the paper [32].

While the translation for derivations for the logic IK is almost trivial in light of the result from Goré and Ramanayake in [25], the main point of interest in Lyon's paper is the employment of the structural refinement method. This allows one to formulate a general class of axiom extensions (here Horn-Scott-Lemmon axioms) for the nested calculus. These rules are being extracted by the propagation rules for the labelled system, which are equivalent to the extension rules while not introducing new relational formulas. To formulate these propagation rules, Lyon relies on some grammar theoretic concepts that help to identify the structure of a nested sequent.

In section 4.2, we formulate a constructive proof translation from derivations of m-NIK to derivations of labIK. In section 4.3, we do something similar in the opposite direction, although the translation, while being effective, requires additional work on the form of derivation trees. We will first introduce two search procedures which will give us derivations that have the desired form for translating them, so we can do the reverse translation in a final step. We omit any translations from extensions of IK, although they should be easy to obtain by a similar method as in [32].

4.2 Translating from m-NIK to labIK

To define the full translation, we will make use of admissible rules in labIK (see figures 3.3.1 and 3.3.2). The broad idea of the translation is that we translate the tree-structure of nested sequents into a tree of relational R formulas in the labelled formalism, as it is usual for such translations. This makes the translation quite easy, as we only consider sequents without any $\leq$-formulas and thus with a lot less structure.

For that we will define a few tools and concepts on the way. Let us first define the operator $\otimes$ (see also [25]), which can be seen as a kind of disjoint union for labelled sequents. It is defined for any two labelled sequents as

$$(\mathcal{R}, \Gamma \Rightarrow \Delta) \otimes (\mathcal{R}', \Sigma \Rightarrow \Pi) \;=\; \mathcal{R}, \mathcal{R}', \Gamma, \Sigma \Rightarrow \Delta, \Pi.$$

We say that a labelled sequent $\mathcal{R}, \Gamma \Rightarrow \Delta$ is a *subsequent* of $\mathcal{R}', \Gamma' \Rightarrow \Delta'$, written $(\mathcal{R}, \Gamma \Rightarrow \Delta) \subseteq (\mathcal{R}', \Gamma' \Rightarrow \Delta')$, if we can write $\mathcal{R}', \Gamma' \Rightarrow \Delta'$ as a sequent union $(\mathcal{R}, \Gamma \Rightarrow \Delta) \otimes (\mathcal{R}'', \Sigma \Rightarrow \Pi)$ for some sequent $\mathcal{R}'', \Sigma \Rightarrow \Pi$.

Another helpful definition for making things more readable are *polarised multisets* $\Gamma^\bullet$ and Γ°, where every formula in the set has the respective polarity (e.g. $\Gamma^\bullet = \{A^\bullet \mid A \in \Gamma\}$). We note that every unnested sequent in m-NIK can be written in terms of two sets of different polarity, as an unnested sequent is nothing more than a multiset of polarised formulas which are either succedent or antecedent formulas. For example, the sequent $p^\bullet, \Box p^\circ, p \supset q^\circ, \neg q^\bullet$ can be divided into two sets $\Gamma^\bullet, \Delta^\circ$ where $\Gamma^\bullet = \{p^\bullet, \neg q^\bullet\}$ and $\Delta^\circ = \{\Box p^\circ, p \supset q^\circ\}$; this is also reminiscent of the sequent formulation $\Gamma \Rightarrow \Delta$.

For the following definition, we use the labelling of a multiset of formulas $\Gamma \subseteq \mathcal{L}^{\Box\Diamond}$ with a label x as $x : \Gamma$, which is defined as the multiset $\{x : A \mid A \in \Gamma\}$ containing all formulas from Γ labelled with x.

Definition 4.1 (From nested to labelled sequents)**.** *For a nested sequent* Γ*, we define its translation into a fully labelled sequent by induction on nesting depth.*

$\mathfrak{L}^x(\Gamma_1^\bullet, \Gamma_2^\circ) = x : \Gamma_1 \Rightarrow x : \Gamma_2$

$\mathfrak{L}^x(\Gamma_1^\bullet, \Gamma_2^\circ, [\Delta_1], ..., [\Delta_n]) = (xRy_1, ..., xRy_n, x : \Gamma_1 \Rightarrow x : \Gamma_2) \otimes (\otimes_{i=1}^n \mathfrak{L}^{y_i}(\Delta_i))$ *where all labels* $x, y_1, ..., y_n$ *are pairwise disjoint.*

Notice that this translation has similarities with definition 3.20, where we already saw that a nested sequent can be read out as a tree. As we will see soon, this will also be the case for their labelled translations. To translate the rules of m-NIK, we would like to represent translations where only a context is known, for which we need the following lemma.

Lemma 4.2. *For any nested sequent* $\Gamma\{\Delta\}$ *with some non-empty context* Δ*, one can always write the labelled translation in the following form.* $\mathfrak{L}^x(\Gamma\{\Delta\}) = \mathfrak{L}^x(\Gamma\{\}) \otimes \mathfrak{L}^y(\Delta)$*. Furthermore, we have* $xR^ny \in \mathfrak{L}^x(\Gamma\{\})$ *with* $depth(\Gamma\{\}) = n$*.*

Proof. By induction on the depth of the context $depth(\Gamma\{\}) = n$.

If $n = 0$, then $\Gamma\{\Delta\} = \Gamma, \Delta$ for some (possibly empty) multiset Γ. Observe then that for an empty context we have $\Gamma\{\} = \Gamma$. So,

$$\begin{aligned}
\mathfrak{L}^x(\Gamma\{\Delta\}) &= \mathfrak{L}^x(\Gamma, \Delta) \\
&= \mathfrak{L}^x(\Gamma_1^\bullet, \Gamma_2^\circ, \Delta_1^\bullet, \Delta_2^\circ) \\
&= x : \Gamma_1, x : \Delta_1 \Rightarrow x : \Gamma_2, x : \Delta_2 \\
&= (x : \Gamma_1 \Rightarrow x : \Gamma_2) \otimes (x : \Delta_1 \Rightarrow x : \Delta_2) \\
&= \mathfrak{L}^x(\Gamma) \otimes \mathfrak{L}^x(\Delta) \\
&= \mathfrak{L}^x(\Gamma\{\}) \oplus \mathfrak{L}^x(\Delta)
\end{aligned}$$

with xR^0x by equivalently $x = x$.

If the sequent Δ occurs inside a nesting of some depth $n > 0$ inside $\Gamma\{\Delta\}$, we can rewrite the sequent such that Δ occurs explicitly nested in the sequent. We do this by first observing that the sequent $[\Delta, \Sigma]$ must occur in $\Gamma\{\Delta\}$ for some (possibly empty) sequent Σ. Second, we can rewrite $\Gamma\{\Delta\}$ into a sequent with $[\Delta, \Sigma]$ as the context; we call it $\Gamma'\{[\Delta, \Sigma]\}$. Because $depth(\Gamma'\{\}) = n-1 < n$, we can use the induction hypothesis: $\mathfrak{L}^x(\Gamma'\{[\Delta, \Sigma]\}) = \mathfrak{L}^x(\Gamma'\{\}) \otimes \mathfrak{L}^y([\Delta, \Sigma])$ with $depth(\Gamma'\{\}) = n - 1$ and $xR^{n-1}y \in \mathfrak{L}^x(\Gamma'\{\})$. By definition, we have $\mathfrak{L}^y([\Delta, \Sigma]) = (yRz \Rightarrow) \otimes \mathfrak{L}^z(\Delta) \otimes \mathfrak{L}^z(\Sigma)$ for some fresh label z. Therefore,

$$\begin{aligned}
\mathfrak{L}^x(\Gamma\{\Delta\}) &= \mathfrak{L}^x(\Gamma'\{[\Delta, \Sigma]\}) \\
&= \mathfrak{L}^x(\Gamma'\{\}) \otimes (yRz \Rightarrow) \otimes \mathfrak{L}^z(\Delta) \otimes \mathfrak{L}^z(\Sigma)
\end{aligned}$$

Reusing the induction hypothesis to gain $\mathfrak{L}^x(\Gamma\{\}) = \mathfrak{L}^x(\Gamma'\{[\Sigma]\}) = \mathfrak{L}^x(\Gamma'\{\}) \otimes (yRz \Rightarrow) \otimes \mathfrak{L}^z(\Sigma)$ we can replace terms to get $\mathfrak{L}^x(\Gamma\{\Delta\}) = \mathfrak{L}^x(\Gamma\{\}) \otimes \mathfrak{L}^z(\Delta)$ with $xR^{n-1}yRz$, so xR^nz and $depth(\Gamma\{\}) = n$. □

Before we continue, we will introduce some notation which will be helpful when working with label substitutions and the following $lift$ lemma. We abbreviate a tuple of labels $(x_1, ..., x_n)$ as $\overline{x}$. We further simplify working with labelled sequents by generally referring to them as $\mathfrak{S}$, therefore also simplifying variable substitution from $\mathcal{R}_{[x/y]}, \Gamma_{[x/y]} \Rightarrow \Delta_{[x/y]}$ to $\mathfrak{S}_{[x/y]}$. We also allow for substituting multiple variables at once, which we will denote as $\mathfrak{S}_{[\overline{x}/\overline{y}]}$ where $\overline{x}$ and $\overline{y}$ have the same size and where every $x_i \in \overline{x}$ substitutes for $y_i \in \overline{y}$. Finally, we also need the concept of output pruning for labelled sequents, which we will denote as $\mathfrak{S}^{\downarrow}$ and is simply defined as removing all succedent formulas from $\mathfrak{S}$.

Another helpful concept is the one of *layers*, which occur in labelled sequents and refer to a set of labels which are only connected through R relational formulas. They can structurally be thought of as the tree-structure of nested sequents, as long as they are also tree-like. The following two definitions have been taken from [23].

Definition 4.3 (Tree-Layer)**.** *For a labelled sequent $\mathcal{R}, \Gamma \Rightarrow \Delta$, let $\mathcal{R}_R$ be the set of relational formulas in $\mathcal{R}$ of the form xRy. We call $\mathcal{R}_R^{\leftrightarrow}$ the reflexive transitive closure of $\mathcal{R}_R \cup \mathcal{R}_R^{-1}$. By $\mathcal{R}_R^{\leftrightarrow}$ being an equivalence relation, we define a layer L to be an equivalence class of $\mathcal{R}_R^{\leftrightarrow}$. We call L a tree-layer, if it constitutes an irreflexive, anti-transitive and rooted tree.*

If there is only a single equivalence class in $\mathcal{R}_R^{\leftrightarrow}$, we also call the sequent $\mathcal{R}, \Gamma \Rightarrow \Delta$ a *single-layer sequent*, or *tree-layer sequent* if $\mathcal{R}_R^{\leftrightarrow}$ is also a tree-layer. The label which is the root of a tree-layer sequent $\mathcal{R}, \Gamma \Rightarrow \Delta$, is called $root(\mathcal{R}, \Gamma \Rightarrow \Delta)$.

Definition 4.4 (Layered sequent)**.** *We say that a labelled sequent $\mathfrak{S}$ is layered iff for any labels x, y, x', y' occurring in $\mathfrak{S}$:*

1. *if $x\mathcal{R}_R^{\leftrightarrow}y$ for $x \neq y$, then $x \nleq y$ and $y \nleq x$; and*
2. *if $x\mathcal{R}_R^{\leftrightarrow}y$, $x'\mathcal{R}_R^{\leftrightarrow}y'$ and $x \leq x'$ for $x \neq x'$, then $y' \nleq y$.*

For layers L_1 and L_2 in $\mathfrak{S}$, we define $L_1 \leq_{\mathfrak{S}} L_2$ whenever there are labels $x \in L_1$ and $y \in L_2$ such that $x \leq y$. A layered sequent consisting only of tree-layers is called tree layered.

We will use the symbol $\mathcal{R}_R^{\leftrightarrow}$, just like R and $\leq$, to refer to a set of ordered pairs in the "structure" of relational labels. We also try to make the distinction between single-layer and layered sequents clear by always hyphenating the former but not the latter.

Proposition 4.5. *For any nested sequent Γ and any label x, $\mathfrak{L}^x(\Gamma)$ is a tree-layer sequent.*

Proof. Follows immediately from the definition of nested sequents and the translation $\mathfrak{L}^x$. □

Proposition 4.6. *Let $\mathcal{T}$ be a derivation of a layered sequent $\mathfrak{S}$ in* labIK*, then every sequent in $\mathcal{T}$ is a layered sequent.*

Proof. Bottom-up induction on the height of $\mathcal{T}$. $\mathfrak{S}$ is layered. The only steps we have to consider are rule applications where we introduce new structure and variables. In general, we only have to consider the set of relational atoms $\mathcal{R}$ and how it changes for these rule applications.

$\supset$R : Assume $\mathcal{R}$ is layered. Then, $\mathcal{R}, x \leq y$ is also layered for a y not occurring in $\mathcal{R}$ because y cannot be R-related to any other label and therefore constitutes the first label of a new layer. There is also no state z s.t. $y \leq z$ and $y \neq z$, so condition 2 of definition 4.4 is fulfilled.

Ref: Follows immediately.

$Trans$: We assume some layered set of relational formulas $\mathcal{R}, x \leq y, y \leq z$ with $x \neq y \neq z$. We can deduce that $x \neq z$, because z must have been freshly introduced at some point in the derivation, where x was already present. Further, we can say that $y \nleq x$, $z \nleq y$ and $z \nleq x$, because the rules of labIK work in such a way that fresh labels can only occur in the second argument of $\leq$ relational formulas. We show that $\mathcal{R}, x \leq y, y \leq z, x \leq z$ is layered. Condition 1 is fulfilled immediately. To check that the added formula of $x \leq z$ does not break condition 2, we have to see that there are no labels u, v with the following property: $x\mathcal{R}_R^{\leftrightarrow}u$, $z\mathcal{R}_R^{\leftrightarrow}v$ and $v \leq u$. This gives some circle $x \leq y \leq z\mathcal{R}_R^{\leftrightarrow}v \leq u\mathcal{R}_R^{\leftrightarrow}x$. And because every fresh label has to be introduced in the second argument of $\leq$ relational formulas, this must be impossible. Thus, such labels u, v existing is impossible.

$\Diamond$L: Let $\mathcal{R}$ be the relation formulas of a layered sequent. Consider $\mathcal{R}, xRy$ with $x \in \mathcal{R}$ and $y \notin \mathcal{R}$. Because y is fresh and there is no new $\leq$-formula, all conditions are fulfilled.

$\Box$R: Apply both arguments of $\supset$R and $\Diamond$L.

F_1: Say, that the relational formulas of the conclusion of F_2, that is $\mathcal{R}, xRy, y \leq z$, is layered.

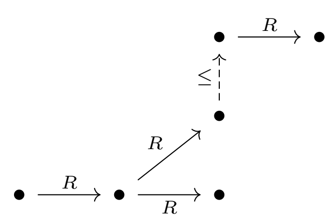
We show that $\mathcal{R}, xRy, y \leq z, x \leq u, uRz$ is layered for a fresh u. As only u is fresh, we only have to check whether its relations to other labels satisfies the condition of definition 4.4. For condition 1, we know that u and z are related through $\mathcal{R}_R^{\leftrightarrow}$ and therefore have to show that z and u are not $\leq$-related. This follows from u being fresh, so no other relational formula with u could have been introduced. For condition 2, we have to consider $x\mathcal{R}_R^{\leftrightarrow}y$, $u\mathcal{R}_R^{\leftrightarrow}z$ and $x \leq u$. Thus, we need to show that $z \nleq y$, which follows from the inductive hypothesis.

F_2: Same as F_1. $\Box$

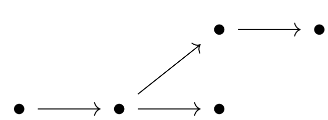
Corollary 4.7. *Any sequents in a* labIK *derivation of a single formula sequent* $\Rightarrow x : A$ *are layered sequents.*

Proof. $\Rightarrow x : A$ is layered by $\mathcal{R} = \emptyset$, so by proposition 4.6 the corollary follows. $\Box$

For translating nested sequent proofs, it is important that we can duplicate layers in a derivation. This is because, for some rules, the labelled calculus creates new layers, while nested sequents simply keep their full structure. For example, after an application of $\Box$R in labIK, the relational set might have the following structure.



In the nested system m-NIK the "same" rule application of $\Box^{\circ}$ would result in a nested sequent which has the following nesting-tree.



Therefore, we would like to be able to copy all the structure we have in layers which are lower with regards to $\leq$ into higher ones. This also includes the "lifting" of antecedent formulas, which is covered by the admissibility of mon_L^g. The rule which we will show is admissible in labIK is defined as follows.

Definition 4.8 (Lift Rule)**.** *For two tree-layer sequents $\mathfrak{S}$ and $\mathfrak{S}'$ who do not share any labels and with x occurring in $\mathfrak{S}$ and y occurring in $\mathfrak{S}'$, we define the following rule.*

$$lift \frac{\mathfrak{S}^{\downarrow} \otimes \mathfrak{S}'_{[x/y]}}{(x \leq y \Rightarrow) \otimes \mathfrak{S} \otimes \mathfrak{S}'}$$

As the next lemma will show, the rule $lift$ is admissible in labIK. As an example, consider the sequent $x \leq y, xRz, yRu, x : A \Rightarrow y : B, z : C$, where we have $\mathfrak{S} = (xRz, x : A \Rightarrow z : C)$ and $\mathfrak{S}' = (\Rightarrow y : B)$. In the following, we show on one side how the lifting rule gets upwards applied to this sequent, while on the other side we have the rule F_2.

$$lift \frac{xRz, xRu, x : A \Rightarrow x : B}{x \leq y, xRz, yRu, x : A \Rightarrow y : B, z : C} \qquad \frac{x \leq y, z \leq z', xRz, yRz', yRu, x : A \Rightarrow y : B, z : C}{x \leq y, xRz, yRu, x : A \Rightarrow y : B, z : C} \, F_2$$

See that the derivability of the sequent in the premise of $lift$ also implies that the sequent obtained from F_2 is derivable. We can show this explicitly by giving the following derivation of admissible rules.

$$\cfrac{\cfrac{\cfrac{xRz, xRu, x : A \Rightarrow x : B}{yRz', yRu, y : A \Rightarrow y : B}\,\text{sub}_V}{x \leq y, z \leq z', xRz, yRz', yRu, x : A, y : A \Rightarrow y : B, z : C}\,\text{wk}}{x \leq y, z \leq z', xRz, yRz', yRu, x : A \Rightarrow y : B, z : C}\,\text{mon}_L^g$$

We write a double line in a derivation tree to signal multiple rule applications of the same rule in a row. Here, it is the rule sub_V, where we substitute y for x and z' for z. By applying the rule F_2 in a final step, we would come back to the sequent we have considered at the start, namely the conclusion of the $lift$ rule.

This shows that, in this specific case, the rule is admissible in labIK. We can use a similar method to show the admissibility of $lift$ in general. Notice that, unlike in our example, most applications of $lift$ will correspond to multiple applications of the confluence rules F_1 and F_2.

Lemma 4.9. *The rule lift is admissible in* labIK*.*

Proof. We assume for the admissibility that the premise sequent is derivable in labIK, namely $\mathfrak{S}^{\downarrow} \otimes \mathfrak{S}'_{[x/y]}$. Define $\overline{v} = (v_1, ..., v_m)$ as the labels occurring in $\mathfrak{S}$, except for x, and $\overline{w} = (w_1, ..., w_m)$ as a tuple of fresh labels w_i. Furthermore, let $\mathcal{R}_R$ be the set of R-formulas in $\mathfrak{S}$. We do the following admissible steps that give us another derivable sequent in the consequent. Except for weakening, where we can introduce all formulas at once, every rule application will happen multiple times in a row. The steps using the general monotonicity rule mon_L^g is the most essential one. It deletes all non-relational formulas in $\mathfrak{S}^{\downarrow}_{[y/x][\overline{w}/\overline{v}]}$, which are all antecedent formulas. They all can be respectively paired up with a formula from $\mathfrak{S}$ s.t. they only have different labels, which are connected via some $\leq$-atom in $\overline{x} \leq \overline{y}$.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathfrak{S}^{\downarrow} \otimes \mathfrak{S}'_{[x/y]}}{\mathfrak{S}^{\downarrow}_{[y/x]} \otimes \mathfrak{S}'}\ \mathrm{sub}_V}{\mathfrak{S}^{\downarrow}_{[y/x][\overline{w}/\overline{v}]} \otimes \mathfrak{S}'}\ \mathrm{sub}_V}{(\overline{v} \leq \overline{w}, x \leq y \Rightarrow) \otimes \mathfrak{S}^{\downarrow}_{[y/x][\overline{w}/\overline{v}]} \otimes \mathfrak{S}^{\downarrow} \otimes \mathfrak{S}'}\ \mathrm{wk}}{(\overline{v} \leq \overline{w}, x \leq y \Rightarrow) \otimes (\mathcal{R}_{[y/x][\overline{w}/\overline{v}]} \Rightarrow) \otimes \mathfrak{S}^{\downarrow} \otimes \mathfrak{S}'}\ \mathrm{mon}^g_L}{(\overline{v} \leq \overline{w}, x \leq y \Rightarrow) \otimes (\mathcal{R}_{[y/x][\overline{w}/\overline{v}]} \Rightarrow) \otimes \mathfrak{S} \otimes \mathfrak{S}'}\ \mathrm{wk}}{(x \leq y \Rightarrow) \otimes \mathfrak{S} \otimes \mathfrak{S}'}\ \mathrm{F}_1, \mathrm{F}_2}{}
$$

The last set of steps consists of applications of the confluence rules, F_1 and F_2, which we can carefully apply: beginning at the root and leafs of the structure $\mathcal{R}_{[y/x][\overline{w}/\overline{v}]}$, which only consists of R relational formulas, we can delete all of the formulas in $\mathcal{R}_{[y/x][\overline{w}/\overline{v}]}$ while retaining $x \leq y$. This will also delete all formulas in $\overline{v} \leq \overline{w}$, giving us a derivation of $(x \leq y \Rightarrow) \otimes \mathfrak{S} \otimes \mathfrak{S}'$, showing that the rule *lift* is admissible. □

Remark 4.10. Notice that we do not require the premise of the *lift* rule to be a tree-layer sequent. This would usually be granted by requiring that the root of the tree-layer sequent $\mathfrak{S}'$ is y, which we can always assume for our purposes as there is no rule directly creating a backwards looking R-atom.

Now, we are ready to proof the essential lemma for the translation.

Lemma 4.11. *Call* labIK$^+$ *the system obtained from* labIK *by adding the admissible rules* mon_L*,* sub_V *and lift. Let*

$$
\frac{\Delta_1, ..., \Delta_n}{\Gamma}
$$

be any rule from m-NIK *with* $n = 0, 1, 2$*. Then, the corresponding rule*

$$
\frac{\mathfrak{L}^x(\Delta_1), ..., \mathfrak{L}^x(\Delta_n)}{\mathfrak{L}^x(\Gamma)}
$$

is derivable in labIK$^+$.

Proof. Ax: $\Gamma = \Gamma\{p^{\bullet}, p^{\circ}\}$
We know from lemma 4.2 that we can write the labelled sequent $\mathfrak{L}^x(\Gamma\{p^{\bullet}, p^{\circ}\})$ as $\mathfrak{L}^x(\Gamma\{\}) \otimes \mathfrak{L}^y(p^{\bullet}, p^{\circ}) = \mathfrak{L}^x(\Gamma\{\}) \otimes (y : p \Rightarrow y : p)$. By the following derivation, it is shown that the translated sequent is derivable.

$$
\dfrac{\dfrac{}{\mathfrak{L}^x(\Gamma\{\}) \otimes (y \leq y, y : p \Rightarrow y : p)}\ \mathrm{Ax}}{\mathfrak{L}^x(\Gamma\{\}) \otimes (y : p \Rightarrow y : p)}\ Ref
$$

$\bot$: $\Gamma = \Gamma\{\bot^\bullet\}$
Write $\mathfrak{L}^x(\Gamma\{\bot^\bullet\}) = \mathfrak{L}^x(\Gamma\{\}) \oplus (y : \bot \Rightarrow)$, which is already an initial sequent in labIK.

$\wedge^\bullet$: $\Gamma = \Gamma\{A \wedge B^\bullet\}$ and $\Delta_1 = \Gamma\{A^\bullet, B^\bullet\}$
We translate these sequents as $\mathfrak{L}^x(\Gamma\{A \wedge B^\bullet\}) = \mathfrak{L}^x(\Gamma\{\}) \otimes (y : A \wedge B \Rightarrow)$ and $\mathfrak{L}^x(\Gamma\{A^\bullet, B^\bullet\}) = \mathfrak{L}^x(\Gamma\{\}) \otimes (y : A, y : B \Rightarrow)$, related through the following derivation.

$$\frac{\mathfrak{L}^x(\Gamma\{\}) \otimes (y : A, y : B \Rightarrow)}{\mathfrak{L}^x(\Gamma\{\}) \otimes (y : A \wedge B \Rightarrow)} \wedge\mathrm{L}$$

$\wedge^\circ$: $\Gamma = \Gamma\{A \wedge B^\circ\}$, $\Delta_1 = \Gamma\{A^\circ\}$ and $\Delta_2 = \Gamma\{B^\circ\}$
With translation by lemma 4.2 we can construct the following derivation.

$$\frac{\mathfrak{L}^x(\Gamma\{\}) \otimes (\Rightarrow y : A) \qquad \mathfrak{L}^x(\Gamma\{\}) \otimes (\Rightarrow y : B)}{\mathfrak{L}^x(\Gamma\{\}) \otimes (\Rightarrow y : A \wedge B)} \wedge\mathrm{R}$$

For $\vee^\bullet$ and $\vee^\circ$, the proofs work the same way as the previous two.

$\supset^\bullet$: $\Gamma = \Gamma\{A \supset B^\bullet\}$, $\Delta_1 = \Gamma\{A \supset B^\bullet, A^\circ\}$ and $\Delta_2 = \Gamma\{A \supset B^\bullet, B^\bullet\}$
The translations are $\mathfrak{L}^x(\Gamma\{A \supset B^\bullet\}) = \mathfrak{L}^x(\Gamma\{\}) \otimes (y : A \supset B \Rightarrow)$, $\mathfrak{L}^x(\Gamma\{A \supset B^\bullet, A^\circ\}) = \mathfrak{L}^x(\Gamma\{\}) \otimes (y : A \supset B \Rightarrow y : A)$ and $\mathfrak{L}^x(\Gamma\{A \supset B^\bullet, B^\bullet\}) = \mathfrak{L}^x(\Gamma\{\}) \otimes (y : A \supset B, y : B \Rightarrow)$. The translated rule can then be derived by the following.

$$\dfrac{\dfrac{\dfrac{\mathfrak{L}^x(\Gamma\{\}) \otimes (y : A \supset B \Rightarrow y : A)}{\mathfrak{L}^x(\Gamma\{\}) \otimes (y \leq y, y : A \supset B \Rightarrow y : A)} \mathrm{wk} \qquad \dfrac{\mathfrak{L}^x(\Gamma\{\}) \otimes (y : A \supset B, y : B \Rightarrow)}{\mathfrak{L}^x(\Gamma\{\}) \otimes (y \leq y, y : A \supset B, y : B \Rightarrow)} \mathrm{wk}}{\mathfrak{L}^x(\Gamma\{\}) \otimes (y \leq y, y : A \supset B \Rightarrow)} \supset\mathrm{L}}{\mathfrak{L}^x(\Gamma\{\}) \otimes (y : A \supset B \Rightarrow)} Ref$$

$\supset^\circ$: $\Gamma = \Gamma\{A \supset B^\circ\}$ and $\Delta_1 = \Gamma^\downarrow\{A^\bullet, B^\circ\}$
The translated sequents are $\mathfrak{L}^x(\Gamma\{A \supset B^\circ\}) = \mathfrak{L}^x(\Gamma\{\}) \otimes (\Rightarrow y : A \supset B)$ and $\mathfrak{L}^x(\Gamma^\downarrow\{A^\bullet, B^\circ\}) = \mathfrak{L}^x(\Gamma^\downarrow\{\}) \otimes (y : A \Rightarrow y : B)$. Observe that $\mathfrak{L}^x(\Gamma^\downarrow\{\})$ only consists of antecedent formulas, meaning we can utilise the lifting from lemma 4.9.

$$\dfrac{\dfrac{\mathfrak{L}^x(\Gamma^\downarrow\{\}) \otimes (y : A \Rightarrow y : B)}{\mathfrak{L}^x(\Gamma\{\}) \otimes (y \leq y' \Rightarrow) \otimes (y' : A \Rightarrow y' : B)} lift}{\mathfrak{L}^x(\Gamma\{\}) \otimes (\Rightarrow y : A \supset B)} \supset\mathrm{R}$$

$\Box^\bullet$: $\Gamma = \Gamma\{\Box A^\bullet, [\Delta]\}$ and $\Delta_1 = \Gamma\{\Box A^\bullet, [\Delta, A^\bullet]\}$
First, observe that $\mathfrak{L}^y(\Box A^\bullet, [\Delta]) = (yRz, y : \Box A \Rightarrow) \otimes \mathfrak{L}^z(\Delta)$. So, we can write the translated sequents as $\mathfrak{L}^x(\Gamma\{\Box A^\bullet, [\Delta, A^\bullet]\}) = \mathfrak{L}^x(\Gamma\{\}) \otimes (yRz, y : \Box A, z : A \Rightarrow) \otimes \mathfrak{L}^z(\Delta)$ and $\mathfrak{L}^x(\Gamma\{\Box A^\bullet, [\Delta]\}) = \mathfrak{L}^x(\Gamma\{\}) \otimes (yRz, y : \Box A \Rightarrow) \otimes \mathfrak{L}^z(\Delta)$.

$$\dfrac{\dfrac{\dfrac{\mathfrak{L}^x(\Gamma\{\}) \otimes (yRz, y : \Box A, z : A \Rightarrow) \otimes \mathfrak{L}^z(\Delta)}{\mathfrak{L}^x(\Gamma\{\}) \otimes (yRz, y \leq y, y : \Box A, z : A \Rightarrow) \otimes \mathfrak{L}^z(\Delta)}\ \text{wk}}{\mathfrak{L}^x(\Gamma\{\}) \otimes (yRz, y \leq y, y : \Box A \Rightarrow) \otimes \mathfrak{L}^z(\Delta)}\ \Box\text{L}}{\mathfrak{L}^x(\Gamma\{\}) \otimes (yRz, y : \Box A \Rightarrow) \otimes \mathfrak{L}^z(\Delta)}\ Ref$$

$\Box^\circ$: $\Gamma = \Gamma\{\Box A^\circ\}$ and $\Delta_1 = \Gamma^\downarrow\{[A^\circ]\}$
Again, using lemma 4.2 and lemma 4.9 we can write $\mathfrak{L}^x(\Gamma\{\Box A^\circ\}) = \mathfrak{L}^x(\Gamma\{\}) \otimes (\Rightarrow y : \Box A)$ and $\mathfrak{L}^x(\Gamma^\downarrow\{[A^\circ]\}) = \mathfrak{L}^x(\Gamma^\downarrow\{\}) \otimes (yRz \Rightarrow z : A)$ and translate the rule as follows. Notice that, by definition of the translation, the variable z cannot occur in $\mathcal{R}$.

$$\dfrac{\dfrac{\mathfrak{L}^x(\Gamma^\downarrow\{\}) \otimes (yRz \Rightarrow z : A)}{\mathfrak{L}^x(\Gamma\{\}) \otimes (y \leq u \Rightarrow) \otimes (uRz \Rightarrow z : A)}\ lift}{\mathfrak{L}^x(\Gamma\{\}) \otimes (\Rightarrow y : \Box A)}\ \Box\text{R}$$

Finally, showing the translation for $\Diamond^\bullet$ and $\Diamond^\circ$ works just like the previous two cases, except that we do not need any application of Ref and $lift$. $\Box$

Theorem 4.12 (From m-NIK to labIK)**.** *For any sequent* Γ*, if* m-NIK $\vdash \Gamma$ *then* labIK $\vdash \mathfrak{L}^x(\Gamma)$*. Furthermore, the translation is effective in that we can constructively translate the full derivation into an admissible derivation for* labIK *using only rules of* labIK$^+$*.*

Proof. Use lemma 4.11 to construct a derivation of $\mathfrak{L}^x(\Gamma)$ in labIK by induction on the height of the proof. Because all rules of labIK$^+$ are admissible in labIK, we have that $\mathfrak{L}^x(\Gamma)$ is also derivable in labIK. $\Box$

Though the translation uses additional admissible rules, we conjecture that all of them would be eliminable if one chooses to translate in a "weaker" way. Firstly, one could translate the premise and conclusion sequents of certain rules under different labels. Secondly, when allowing to weaken the sequent already in the process of translating, we might not have to rely on weakening. Both these steps rely on making $\mathfrak{L}^x$ more versatile in that the choice of labels and weakened formulas depends heavily on which rules are used on this sequent as well as from which ones it was obtained. Instead, we could also include the admissible rules of wk, sub_V and $lift$ as primitive rules of labIK to gain a more satisfying proof translation.

4.3 Translating from labIK to m-NIK

This section presents a translation from labIK, the labelled sequent calculus presented in section 3.3.1, and m-NIK, which was presented in section 3.3.2. We will do the translation in two major steps. At first, in subsection 4.3.1, we will define a proof search procedure that yields sequents of a certain form, which we can translate into the nested formalism. The trouble with considering arbitrary derivations in labIK for translating is that we have a lot more structure in sequents, and the proof system labIK is also fully invertible, unlike m-NIK. This makes only certain rule applications possible to translate properly. In a second step, subsection 4.3.2, we will define the translation function for these specific sequents obtained from the algorithm, and show that we can translate the rule into their respective counterpart of m-NIK.

4.3.1 Algorithm for Linear Proofs

Let us first acknowledge the fact that bi-labelled sequents, which have two types of relations, are generally not translatable into nested sequents. The latter only internalises the R-relation. Therefore, bi-labelled sequents carry more information than nested ones. To reduce this information, we only want to look at the topmost layer of the labelled sequent. This requires a labIK derivation to have a certain form, if we want to translate it. One might either try to modify an existing tree such that it displays the desired form. Alternatively, we can impose a certain proof search procedure that will give us such a derivation. The latter approach is less constructive as it does not translate derivations directly and therefore does not allow for any transferal of structural properties. At the same time, this approach is simpler to handle as it only requires a sophisticated proof search algorithm, instead of constructing a working procedure to edit an already-established derivation tree. Here, we will focus on creating a specific derivation tree in labIK which has the right form to be translated. To sketch the problem at hand, let us consider the following example.

Example 4.13. Consider the derivation of the sequent $x : \Box(p \wedge q) \Rightarrow x : \Box p, x : q \supset p$ in labIK.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x \leq y, x \leq z, zRu, u \leq u, x : \Box(p \wedge q), u : p, u : q, y : q \Rightarrow u : p, y : p}\,\text{Ax}}{x \leq y, x \leq z, zRu, x : \Box(p \wedge q), u : p, u : q, y : q \Rightarrow u : p, y : p}\,Ref}{x \leq y, x \leq z, zRu, x : \Box(p \wedge q), u : p \wedge q, y : q \Rightarrow u : p, y : p}\,\wedge\text{L}}{x \leq y, x \leq z, zRu, x : \Box(p \wedge q), y : q \Rightarrow u : p, y : p}\,\Box\text{L}}{x \leq y, x : \Box(p \wedge q), y : q \Rightarrow x : \Box p, y : p}\,\Box\text{R}}{x : \Box(p \wedge q) \Rightarrow x : \Box p, x : q \supset p}\,\supset\text{R}$$

Observe that the derivation does not rely on the application of $\supset$R, as the initial sequent is formed from states that do not rely on y. Similarly, if we would try to prove the corresponding sequent in the nested calculus m-NIK, i.e. $\Box(p \wedge q)^\bullet, \Box p^\circ, q \supset p^\circ$, we would not gain a valid proof tree if we applied $\supset^\circ$ first. So, we would have to backtrack to apply $\Box$R instead.

As we will see, this translatability corresponds to labelled sequents keeping a certain form, which we will call *linearly layered*. By proposition 4.6 and corollary 4.7, we already know that any sequent that occurs in a derivation of a sequent of the form $\Rightarrow x : A$ is layered. We also know from proposition 6.7 in [24] that the relation $\leq_{\mathfrak{S}}$ between layers is always tree-like. Now, we want to restrict this further so that $\leq_{\mathfrak{S}}$ is linear.

Definition 4.14 (Linearly layered sequents)**.** *A labelled sequent $\mathfrak{S}$ is linearly layered, if it is tree layered and the relation $\leq_{\mathfrak{S}}$ is non-branching (i.e. there are no layers L_1, L_2, L_3 s.t. $L_1 \leq L_2$, $L_1 \leq_{\mathfrak{S}} L_3$, $L_2 \not\leq_{\mathfrak{S}} L_3$ and $L_3 \not\leq_{\mathfrak{S}} L_2$).*

If we have a linearly layered sequents, we are able to find $\leq_{\mathfrak{S}}$-maximal (as well as a minimal) layers in the sequent. We call L_{max} (or $L_{max}(\mathfrak{S})$ if we want to make it explicit which sequent it is the maximum layer of) the set of labels that make up this layer. We will establish how to obtain linear layeredness by working towards a proof search algorithm. At first, we will introduce some general notions that are important to define the algorithm, then we introduce a primitive algorithm that helps us proving that the final algorithm will actually terminate.

We rely on ideas from [23] for the proof search algorithm. Just like most proof search algorithms, implementing it directly will usually exhaust the whole search space and therefore might create sequents that are non-linear and thus untranslatable. For instance, multiple applications of $\supset$R and $\Box$R in the same layer will create some branching in the structure of $\leq_{\mathfrak{S}}$. To prevent such a thing from happening, we shall rely on the single succession property (lemma 4.24) for labIK, which will allow us to only choose one application of multiple possible $\supset$R and $\Box$R on the same layer. To be able to make this choice, a sequent has to be what we will call *saturated*. This property is also called *almost happy* in [23], as almost all formulas have been analysed in such a sequent.

To make the notation easier for the next definition, we also use *polarised formulas for labelled sequents*. It is clear where relational formulas appear in a labelled sequent $\mathfrak{S} = (\mathcal{R}, \Gamma \Rightarrow \Delta)$, such that we can just write $xRy \in \mathfrak{S}$, meaning $xRy \in \mathcal{R}$. We would like to do the same for the other labelled formulas, without having the need to refer to the antecedent or succedent set directly. We therefore define $x : A^\circ$ as the formula $x : A$ occurring in the succedent of a sequent and respectively $x : A^\bullet$ as the formula $x : A$ occurring in the antecedent.

Definition 4.15 (Saturated Sequents)**.** *We call a sequent $\mathfrak{S}$ saturated, if it satisfies the following conditions for any labels x, y, z occurring in $\mathfrak{S}$:*

1. *The following formulas are not contained in $\mathfrak{S}$: $x : A \wedge B^\circ$, $x : A \vee B^\circ$, $x : A \wedge B^\bullet$, $x : A \vee B^\bullet$ and $x : \Diamond A^\bullet$.*
2. *If $x : A \supset B^\bullet \in \mathfrak{S}$, then for each $x \leq y \in \mathfrak{S}$ it holds that either $y : A^\circ \in \mathfrak{S}$ or $y : B^\bullet \in \mathfrak{S}$.*
3. *If $x : \Box A^\bullet \in \mathfrak{S}$, then for each $x \leq y, yRz \in \mathfrak{S}$ it holds that $z : A^\bullet \in \mathfrak{S}$.*
4. *If $x : \Diamond A^\circ \in \mathfrak{S}$, then for each $xRy \in \mathfrak{S}$ it holds that $y : A^\circ \in \mathfrak{S}$.*

To incorporate saturation as a part of our proof search procedure, we introduce the notion of saturation tree.

Definition 4.16 (Saturation Tree)**.** *The saturation tree $\mathcal{T}$ of a sequent $\mathfrak{S}$ is a derivation tree built only of the rules $\wedge$L, $\wedge$R, $\vee$L, $\vee$R, $\supset$L, $\Box$L, $\Diamond$L and $\Diamond$R, where all leaves are saturated and $\mathfrak{S}$ is the root of $\mathcal{T}$. We call the leaves of $\mathcal{T}$ saturations of $\mathfrak{S}$.*

See that the saturation tree always exists for any labelled sequent, as we can satisfy all conditions of sequent saturation by applying respective rules. See also that, except for $\Diamond$L, all rule applications of saturation will not introduce new labels. The process of saturation will always terminate, as we only introduce finitely many new labels via $\Diamond$L. All other rules analyse formulas. Rules which can be applied multiple times, $\supset$L, $\Box$L and $\Diamond$R, are also only applied a finite amount of times, as there are only finitely many labels. Finally, as there can only be finitely many relational formulas xRy, condition 4 of definition 4.15 will be fulfilled after finitely many applications of $\Diamond$R.

One of the essential steps for our algorithm and for the ability to define the sequent translation in the first place will the lifting of a labelled sequent (see [23]). Unlike the previously introduced lift rule, this lifting is defined for any sequent and only uses the primitive rules of labIK.

Definition 4.17 (Lifting)**.** *Let $\mathfrak{S}$ be a saturated tree layered sequent, and let $L = \{y_1, ..., y_n\}$ be a layer in $\mathfrak{S}$ with $x \in L$. Let $\hat{L} = \{\hat{y}_1, ..., \hat{y}_n\}$ be a fresh set of labels. We define $\mathfrak{S}{\uparrow^x}$ to be the sequent containing for every $i \in \{1, ..., n\}$:*

1. *relational atom $\hat{y}_i \leq \hat{y}_i$;*
2. *for every label w occurring in $\mathfrak{S}$: the relational atom $w \leq \hat{y}_i$ whenever $w \leq y_i \in \mathfrak{S}$;*

3. for every $i' \in \{1, ..., n\}$: the relational atom $\hat{y}_i R \hat{y}_{i'}$ whenever $y_i R y_{i'}$.

For some formula $x : F^\circ \in \mathfrak{S}$ with $F = A \supset B$ or $F = \Box A$, we define the lifting $\mathfrak{S}{\uparrow}^{x:F^\circ}$ as follows:

- *If $F = A \supset B$, then $\mathfrak{S}{\uparrow}^{x:A \supset B^\circ} = \mathfrak{S}{\uparrow}^{x} \otimes (\hat{x} : A \Rightarrow \hat{x} : B)$ with $x \leq \hat{x}$.*
- *If $F = \Box A$, then $\mathfrak{S}{\uparrow}^{x:\Box A^\circ} = \mathfrak{S}{\uparrow}^{x} \otimes (\hat{x} R z, z \leq z \Rightarrow \hat{z} : A)$ with $x \leq \hat{x}$ and z fresh.*

Note that, in principle, lifting can also be applied to any labelled sequent; for example, sequents that are neither saturated nor tree layered. Just like for saturation, we can obtain the *lifting tree* of a sequent, which we will write as a single rule application.

Definition 4.18 (Lifting Rule)**.** *The lifting rule for a formula $x : F$ with $F = A \supset B$ or $F = \Box A$ is defined as follows.*

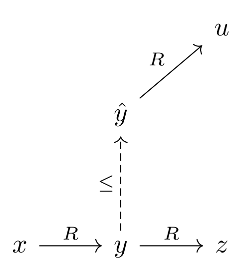
$$\frac{\mathfrak{S} \otimes \mathfrak{S}{\uparrow}^{x:F}}{\mathfrak{S}}\ lifting_{x:F}$$

We note that the $lifting$ rule is admissible in labIK, by construction. The rule can also be seen as a generalisation of the rule $lift$ (see definition 4.8) together with an application of weakening and either $\supset$R ($F = A \supset B$) or $\Box$R ($F = \Box A$).

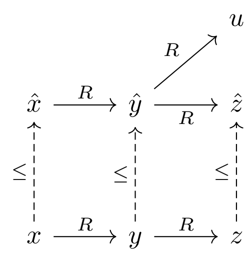
Example 4.19. Consider the following (relationally saturated) sequent $xRy, yRz, x \leq x, y \leq y, z \leq z \Rightarrow y : \Box A$ whose relational structure we can represent as follows (we omit reflexive relational formulas).

$$x \xrightarrow{R} y \xrightarrow{R} z$$

Let us apply the lifting rule to the sequent with $F = \Box A$. First, we essentially apply the rule $\Box$R (also with Ref) to get the sequent $xRy, yRz, x \leq x, y \leq y, z \leq z, y \leq \hat{y}, \hat{y}Ru, u \leq u \Rightarrow u : A$ with its relational structure:



Now, we simply compute the lifting over y, which can be done in a proof by using the relational rules. The lifting itself gives us the sequent $\hat{x} \leq \hat{x}, \hat{y} \leq \hat{y}, \hat{z} \leq \hat{z}, x \leq \hat{x}, y \leq \hat{y}, z \leq \hat{z}, \hat{x}R\hat{y}, \hat{y}R\hat{z} \Rightarrow$. So, together we obtain the sequent $xRy, yRz, y \leq \hat{y}, \hat{y}Ru, u \leq u, x \leq x, y \leq y, z \leq z\hat{x} \leq \hat{x}, \hat{y} \leq \hat{y}, \hat{z} \leq \hat{z}, x \leq \hat{x}, y \leq \hat{y}, z \leq \hat{z}, \hat{x}R\hat{y}, \hat{y}R\hat{z} \Rightarrow u : A$ with the relational structure:



Notice that the non-reflexive $\leq$ formulas $(x \leq \hat{x}, y \leq \hat{y}, z \leq \hat{z})$ come from condition 2 of definition 4.17 and the fact that we already had reflexive $\leq$ formulas $x \leq x, y \leq y$ and $z \leq z$.

We rely on the notion of *redundant rules*. These are rule applications in a derivation tree which do not yield any new information and can be deleted in principle. They are defined as follows.

Definition 4.20 (Redundant Rules)**.** *A rule application in labIK*

$$\frac{\mathfrak{S}_1, ..., \mathfrak{S}_n}{\mathfrak{S}}$$

with $n = 1, 2$ is redundant, either if it introduces a labelled formula in some $\mathfrak{S}_i$ that was already in $\mathfrak{S}$, or

- the rule is an instance of F_1 and it introduces some $x \leq u, uRz$ in $\mathfrak{S}_1$ while there are already some $x \leq w, wRz$ in $\mathfrak{S}$; or

- the rule is an instance of F_2 and it introduces some $y \leq u, zRu$ in $\mathfrak{S}_1$ while there are already some $y \leq v, zRv$ in $\mathfrak{S}$.

Besides the previously introduced formula saturation tree we also include *relationally saturated sequent* solely for the purpose of the algorithm.

Definition 4.21 (Relationally Saturated Sequent)**.** *A sequent $\mathfrak{S}$ is relationally saturated if it is saturated and satisfies the following conditions for any labels x, y, z occurring in $\mathfrak{S}$:*

1. *$x \leq x \in \mathfrak{S}$.*
2. *If $x \leq y, y \leq z \in \mathfrak{S}$ then $x \leq z \in \mathfrak{S}$.*
3. *If $xRy, y \leq z \in \mathfrak{S}$ then $x \leq u, uRz \in \mathfrak{S}$ for some label u.*
4. *If $xRy, x \leq z \in \mathfrak{S}$ then $y \leq u, zRu \in \mathfrak{S}$ for some label u.*

Just like saturation, we can find a canonical way of defining a way to obtain a relationally saturated sequent.

Definition 4.22 (Relational Saturation Tree)**.** *A relational saturation tree of a sequent is obtained by iteratively applying non-redundant relational rules (Ref, $Trans$, F_1 and F_2) and computing the saturation tree.*

0. Given a sequent $\Rightarrow x : A$ derivable in labIK, define $\mathcal{T}_0$ as the derivation tree consisting only of the node $x \leq x \Rightarrow x : A$.
1. Relationally saturate the leaves of $\mathcal{T}_i$ by computing their relational saturation trees.
2. If all leaves of $\mathcal{T}_i$ are initial sequents, terminate.
   $\rightarrow$ A proof of $\Rightarrow x : A$ is obtained.
3. Otherwise, pick a non-axiomatic leaf sequent $\mathfrak{S}'$ in $\mathcal{T}_i$. Apply any non-redundant rule ($\supset$R or $\Box$R) to $\mathfrak{S}'$, and go back to step 1 ($i \mapsto i + 1$).

Figure 4.3.1: Proof search algorithm for labIK

See that the process of relational saturation will also terminate eventually, as we still do not introduce new layers and all layers stay finite, so there is only a finite amount of rule applications. This allows us to define a simple proof search procedure for labIK (see figure 4.3.1), to show that it is enough to rely on linearly layered sequents.

As we can see, the algorithm does not include "all" possible rule applications, as we could, for instance, apply the rules of $\Box$L or F_2 multiple times. All of these applications that are not included in the proof search procedure will be *redundant*, though. Except for F_1 and F_2, all rules will introduce formulas that already existed in the sequent, and therefore are unnecessary. The confluence rules become redundant after some $lifting$ because they have to introduce a fresh label that has the exact same properties as another label that was introduced by $lifting$.

Lemma 4.23. *The algorithm presented in figure 4.3.1 terminates for a sequent $\Rightarrow x : A$ derivable in* labIK.

Proof. We can assume some derivation $\mathcal{T}$ of the sequent $\Rightarrow x : A$. We can observe that the redundant rule applications in $\mathcal{T}$ can always be eliminated from a derivation. This is simply because any redundant F_1 and F_2 rule will introduce a label that has the exact same properties as an already existing one. All other redundant rules can be eliminated by definition and the fact that contraction is admissible.

By limiting the search algorithm to non-redundant formulas, there will always be finite applications of those. Thus, eventually the algorithm includes all non-redundant rule applications of $\mathcal{T}$. It therefore creates a derivation that covers all necessary rules of $\mathcal{T}$, which must be valid by rule invertibility (lemma 3.17). $\Box$

This algorithm is merely a first step towards an algorithm that will only yield linearly labelled sequents. To show that this is possible, we show that it is enough to consider only one succedent formula in a saturated sequent. The following lemma is similar to theorem 14 from [43], where a multiset of nestings is shown to be derivable in a basic nested system iff some of these nestings is derivable in the same system. This theorem also sparked also the idea for this lemma.

Lemma 4.24 (Single Succession)**.** *Let $\mathcal{R}, \Gamma \Rightarrow \Delta$ be a relationally saturated sequent appearing in the proof search described in figure 4.3.2, then* labIK $\vdash \mathcal{R}, \Gamma \Rightarrow \Delta$ *iff* labIK $\vdash \mathcal{R}, \Gamma \Rightarrow x : C$ *for some $x : C \in \Delta$. We call $x : C$ the single succedent of the sequent.*

Proof. ($\Leftarrow$) Follows trivially from the admissibility of weakening.

($\Rightarrow$) We prove it by induction on the height of the derivation of $\mathcal{R}, \Gamma \Rightarrow \Delta$ under the algorithm from figure 4.3.1. If it is an initial (relationally saturated) sequent with $y : p$ as its principal succedent formula, take $y : p$ to be $x : C$. If the initial sequent is obtained from some formula $u : \bot$ in the antecedent, let $x : C$ be arbitrary. As we are considering a relationally saturated sequent, we can assume that the sequent we are working with was obtained after step 1, so that we are in an instance of step 3. Thus, we only need to consider applications of $\supset$R and $\Box$R. After which we need to saturate the sequent to obtain again a saturated sequent allowing us to use the inductive hypothesis. Especially checking that the rule $\supset$L does not cause any issues is important, as it can introduce new succedent formulas from already existing ones.

Consider $\supset$R as a rule application in step 3. So, assume a derivable and relationally saturated sequent $\mathcal{R}, \Gamma \Rightarrow \Delta, x : A \supset B$ s.t. the previous step was $\supset$R with $x : A \supset B$ principal. Then, $\mathcal{R}, x \leq y, y : A, \Gamma \Rightarrow \Delta, y : B$ is provable with y not being in the original sequent.

In the algorithm, the sequent $\mathcal{R}, x \leq y, y : A, \Gamma \Rightarrow \Delta, y : B$ gets immediately saturated. In the saturation, only those labels get introduced which are in the same layer as y (by $\Diamond$L), thus no new layers are created in the process. We can therefore write any saturation of $\mathcal{R}, x \leq y, y : A, \Gamma \Rightarrow \Delta, y : B$ as $(\mathcal{R}, x \leq y, \Gamma \Rightarrow \Delta) \otimes \mathfrak{S}_i$ with $\mathfrak{S}_i$ being a tree-layer sequent with $root(\mathfrak{S}_i) = y$. This gives us the following derivation tree.

$$\dfrac{\begin{array}{c}(\mathcal{R}, x \leq y, \Gamma \Rightarrow \Delta) \otimes \mathfrak{S}_1 \quad \ldots \quad (\mathcal{R}, x \leq y, \Gamma \Rightarrow \Delta) \otimes \mathfrak{S}_n \\ \vdots\, \mathcal{S} \\ \mathcal{R}, x \leq y, y : A, \Gamma \Rightarrow \Delta, y : B\end{array}}{\mathcal{R}, \Gamma \Rightarrow \Delta, x : A \supset B} \supset\text{R}$$

By induction hypothesis every $(\mathcal{R}, x \leq y, \Gamma \Rightarrow \Delta) \otimes \mathfrak{S}_i$ is also derivable with some labelled formula $z_i : C_i$ occurring either in Δ or in the succedent of $\mathfrak{S}_i$. We consider two cases: Either all $z_i : C_i$ occur in $\mathfrak{S}_i$ or there is some saturation, where $z_i : C_i \in \Delta$.

Consider $z_i : C_i \in \Delta$ for some saturation $(\mathcal{R}, x \leq y, \Gamma \Rightarrow \Delta) \otimes \mathfrak{S}_i$. We first cover the case of $(\mathcal{R}, x \leq y, \Gamma \Rightarrow \Delta) \otimes \mathfrak{S}_i$ being obtained by the rule $\bot$ with the principal formula $u : \bot^\bullet \in \mathfrak{S}_i$ ($u : \bot^\bullet \in \Gamma$ is impossible as $\mathcal{R}, \Gamma \Rightarrow \Delta, x : A \supset B$ would have been already an initial sequent). Then, we can choose a different single succedent from $\mathfrak{S}_i$, as it is an arbitrary choice in this case. Note, that there must be at least some succedent formula in $\mathfrak{S}_i$, as we have the succedent formula $y : B$ at the root of $\mathcal{S}$ and every upwards rule application that starts with a succedent formula must also end with at least one succedent formula.

For the case that both $z_i : C_i \in \Delta$ and $(\mathcal{R}, x \leq y, \Gamma \Rightarrow \Delta) \otimes \mathfrak{S}_i$ was not obtained by $\bot$, we claim that the sequent $\mathcal{R}, \Gamma \Rightarrow z_i : C_i$ is provable. This follows from the observation that the layer of y (i.e. the maximum layer of the sequent, which is also the layer of $\mathfrak{S}_i$) is not needed to generate an initial sequent, if $(\mathcal{R}, x \leq y, \Gamma \Rightarrow z_i : C_i) \otimes \mathfrak{S}_i^{\downarrow}$ is provable. This follows from Δ already being saturated and thus, to make an axiomatic sequent, $z_i : C_i$ must create a new layer, which branches away from the layer of y.

Consider $z_i : C_i^\circ \in \mathfrak{S}_i$ for all $i = 1, ..., n$. Then, the sequent $(\mathcal{R}, x \leq y, \Gamma \Rightarrow z_i : C_i) \otimes \mathfrak{S}_i^{\downarrow}$ is derivable for every i. Note that formulas in Δ are unnecessary for the saturation tree $\mathcal{S}$, as the only right side rule that repeats the principal formula is $\Diamond$R, which cannot be applied as only new R formulas will be introduced in the fresh maximum layer. (This does not hold for antecedent rules.) By weakening and $z_i : C_i^\circ \in \mathfrak{S}_i$, the sequent $(\mathcal{R}, x \leq y, \Gamma \Rightarrow) \otimes \mathfrak{S}_i$ is also derivable for every i.

Inputting these sequents instead of $(\mathcal{R}, x \leq y, \Gamma \Rightarrow \Delta) \otimes \mathfrak{S}_i$ (i.e. deleting Δ) will yield the following valid derivation.

$$\dfrac{\begin{array}{c}(\mathcal{R}, x \leq y, \Gamma \Rightarrow) \otimes \mathfrak{S}_i \quad \dots \quad (\mathcal{R}, x \leq y, \Gamma \Rightarrow) \otimes \mathfrak{S}_n \\ \vdots \mathcal{S} \\ \mathcal{R}, x \leq y, y : A, \Gamma \Rightarrow y : B\end{array}}{\mathcal{R}, \Gamma \Rightarrow x : A \supset B} \supset\text{R}$$

This provides us with a sequent with $x : A \supset B$ as the single succedent.

The case for the application of $\Box$R works similar to the previous one.

□

Remark 4.25. As follows from the proof, the single succedent will always be a formula of the form $x : A \supset B$ or $x : \Box A$ if it is not an initial sequent.

This result about derivations in labIK tells us that it is enough to consider derivations of a certain form, which only consists linearly layered sequents. We call such a derivation a *linear proof.*

Corollary 4.26. *Every provable sequent* $\Rightarrow x : A$ *in* labIK *has a linear proof.*

Proof. Consider the derivation tree of $\Rightarrow x : A$ obtained from the algorithm presented in figure 4.3.1. By lemma 4.24, after step 1 (relational saturation of all leaf sequents) we can always choose one succedent formula for each leaf sequent, so that we are allowed to remove all other succedent formulas. We only have to show that when new layers get introduced in the proof, we can prevent a branching of layers. Taking $\Box$R as an example ($\supset$R works similarly), suppose that for some saturated and derivable sequent $\mathcal{R}, \Gamma \Rightarrow \Delta, x : \Box A$, the sequent $\mathcal{R}, \Gamma \Rightarrow x : \Box A$ is derivable, otherwise applying $\Box$R is unnecessary by the single succedent lemma. This gives us the sequent $\mathcal{R}, x \leq y, yRz, \Gamma \Rightarrow z : A$. Saturating this sequent will only yield new formulas that are labelled with either y or z as all other labels have been introduced before and were already saturated. Applying non-redundant relational rules will also only introduce new labels in the new layer. Therefore, after the relational saturation process, all succedent formulas are labelled with labels from the same layer as y and z. So, when again in step 3 and applying $\Box$R (resp. $\supset$R), the single succedent must be in the newly created layer.

Starting with $\Rightarrow x : A$, i.e. x being in the maximum layer and $x : A$ as a single succedent, all $\Box$R (and $\supset$R) will be applied to formulas in the maximum layer, thus always creating a linear, i.e. non-branching, $\leq_{\mathfrak{S}}$ structure between layers. □

To properly construct such a linear derivation, we again use lifting and saturation to define another proof search algorithm (see figure 4.3.2). Just like the previous algorithm, we rely on knowing that the sequent we start with is derivable. For a proper decision procedure for labIK and labIS4 that can also find countermodels, we refer the reader to [23, 24]. The reason why it is difficult to have a terminating procedure, also for false formulas, is due to the fact that we might end up in certain loops. As mentioned previously, having full invertibility comes with the trade-off of keeping a lot of formulas around, such that one might apply them as often as one likes. These loops might not only give us actual duplicates of formulas, but might just create new labels that are essentially the same as already established one (just like with F_1 and F_2 having a different redundancy condition). This might happen indefinitely for unprovable formulas, such that one needs to incorporate a loop checking procedure that recognises when the formulas of fresh labels are equivalent to already obtained formulas.

0. Given a sequent $\Rightarrow x : A$ derivable in labIK, define $\mathcal{T}_0$ as the derivation tree consisting only of the node $x \leq x \Rightarrow x : A$.
1. Saturate the leaves of $\mathcal{T}_i$ by computing the saturation trees.
2. If all leaves of $\mathcal{T}_i$ are initial sequents, terminate.

   $\rightarrow$ A linear proof of $\Rightarrow x : A$ is obtained.
3. Otherwise, pick a non-axiomatic leaf sequent $\mathfrak{S}'$ in $\mathcal{T}_i$.
   (a) If there is some succedent formula $x : F$ for $F = A \supset B$ or $F = \Box A$ in $\mathfrak{S}'$ s.t. $x \in L_{max}(\mathfrak{S}')$, choose some formula $x : F$ with $x \in L_{max}(\mathfrak{S}')$. Compute the lifting $\mathfrak{S} \otimes \mathfrak{S}\uparrow^{x:F}$ and go back to Step 1 ($i \mapsto i + 1$).
   (b) Otherwise, backtrack to the last instance of Step 3a where one can choose a different formula $x : F$ that has not been tried before, and choose this formula instead.

Figure 4.3.2: Linear proof search algorithm for labIK

As we have seen in corollary 4.26 it can be enough to pick a succedent formula $x : A \supset B$ or $x : \Box A$ that is labelled by a variable from the maximum layer of the sequent. This is also captured by how the algorithm works and it essentially emulates the behaviour of the respective rules $\supset^\circ$ and $\Box^\circ$ from the nested calculus m-NIK.

Proposition 4.27. *For any sequent $\Rightarrow A$ derivable in labIK, the algorithm presented in figure 4.3.2 always terminates.*

Proof. Because we assume that the initial formula A is provable, there exists some finite number of rule applications that constitute a valid derivation of the formula, i.e. the sequent $\Rightarrow x : A$. Observe that due to saturation, the rules $\wedge$L, $\wedge$R, $\vee$L, $\vee$R and $\Diamond$L can be applied as much as possible. Further, the rules $\supset$L, Ref, $Trans$, $\Box$L, $\Diamond$R, F_1 and F_2 also get applied as long as they do not become redundant. By corollary 4.26 it is enough to only include applications of $\supset$R and $\Box$R on the maximum layer, so the algorithm covers all necessary applications of $\supset$R and $\Box$R as well. Leaving out these unnecessary occurrences of $\supset$R and $\Box$R as well as rule applications that become impossible without them (i.e. rule applications relying on subformulas or variables introduced by those rules), all rules from the original derivation occur in the derivation constructed by the algorithm. This follows from invertibility of all rules. Therefore, any useful rule application from the original proof will eventually occur in the derivation obtained by the algorithm. $\Box$

Remark 4.28. The final derivation we obtain from the algorithm might have more rule applications than necessary, due to saturation and lifting exhausting almost all of the search space.

By construction, the linear algorithm forms only linearly layered sequents, which can be translated into simple nested ones.

When showing the termination of the proof search algorithm, we made use of the fact that we can ignore certain rule applications (especially $\supset$R and $\Box$R rules) by using lemma 4.24. It seems as if it should also be possible to obtain a derivation satisfying our condition for linear sequents directly by deleting the unnecessary rule applications. The issue with this approach is that it might still need some careful tinkering with the original derivation, as we might end up in a situation where we have two independent applications of $\supset$R or $\Box$R before applying a branching (i.e. two premise) rule, such that we require one application of $\supset$R (or $\Box$R) for one branch and the other application for the other branch. Consider, for instance, the following simple example.

Example 4.29. Consider the following derivation of the formula $(\Box p \vee \Box q) \supset (\Box p \vee \Box q)$ in labIK with $\mathcal{R} = \{x \leq y, y \leq z, zRu, y \leq z', z'Ru'\}$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{R}, u' \leq u', y : \Box p, u' : q \Rightarrow u : p, u' : q}\,\text{Ax}}{\mathcal{R}, y : \Box q, u' : q \Rightarrow u : p, u' : q}\,Ref}{\mathcal{R}, y : \Box q \Rightarrow u : p, u' : q}\,\Box\text{L} \qquad \dfrac{\dfrac{\dfrac{}{\mathcal{R}, u \leq u, y : \Box p, u : p \Rightarrow u : p, u' : q}\,\text{Ax}}{\mathcal{R}, y : \Box p, u : p \Rightarrow u : p, u' : q}\,Ref}{\mathcal{R}, y : \Box p \Rightarrow u : p, u' : q}\,\Box\text{L}}{x \leq y, y \leq z, zRu, y \leq z', z'Ru', y : \Box p \vee \Box q \Rightarrow u : p, u' : q}\,\vee\text{L}}{x \leq y, y \leq z, zRu, y : \Box p \vee \Box q \Rightarrow u : p, y : \Box q}\,\Box\text{R}}{x \leq y, y : \Box p \vee \Box q \Rightarrow y : \Box p, y : \Box q}\,\Box\text{R}}{x \leq y, y : \Box p \vee \Box q \Rightarrow y : \Box p \vee \Box q}\,\vee\text{R}}{\Rightarrow x : (\Box p \vee \Box q) \supset (\Box p \vee \Box q)}\,\supset\text{R}$$

As one can see, if we tried making any of the axiomatic sequents in the derivation linearly layered, we would have to remove one application of the $\Box$R rule, which in turn would make the proof invalid because the other leaf would be deleted with it. Instead, one could think of moving these unnecessary rule applications into a later point of the branch where they are necessary. In this case, we could postpone the first application of $\Box$R for after the branching rule $\vee$L and only apply it to the right branch. A similar procedure could be done to the other application of $\Box$R, giving us a derivation which is actually translatable into an m-NIK derivation.

Another approach that does not rely on a proof search algorithm would be to saturate every sequent in an already given derivation tree. In the previous example, this would prevent applying $\Box$R two times in a row. The disadvantage this approach might have is that, just like with the proof search algorithms, we generally have much more rule application in our derivation than necessary.

4.3.2 Translation for Linearly Layered Sequents

To define the translation function that translates sequents of labIK into nested sequents, we first consider only translating labelled sequents that do not contain any formula $x \leq y$ and are therefore tree-layer sequents. We rely here on the translation function for classical tree-labelled and tree-hypersequents from [25].

Definition 4.30 (Immediate Subtree)**.** *Let $\mathfrak{S}$ be a tree-layered sequent with $root(\mathfrak{S})$ as its root. An immediate subtree sequent $\mathfrak{S}'$ of $\mathfrak{S}$ is defined as follows:*

1. $\mathfrak{S}' \subseteq \mathfrak{S}$,
2. *if $y = root(\mathfrak{S}')$ and $x = root(\mathfrak{S})$, then $xRy \in \mathfrak{S}$,*
3. *If a label x occurs in $\mathfrak{S}'$ and $xRy \in \mathfrak{S}$, then y occurs in $\mathfrak{S}'$, and*
4. *For every label x occurring in $\mathfrak{S}'$, all formulas labelled with x in $\mathfrak{S}$ must also occur in $\mathfrak{S}'$.*

The set of immediate subtree sequents of a tree-layer sequent $\mathfrak{S}$ is called $subtr(\mathfrak{S})$.

To consider *subtrees*, as opposed to immediate subtrees, we relax condition 2 of the definition to there being some path R^n from the root of the main sequent to the root of the subtree. Intuitively, a subtree of a sequent corresponds to a downwards closed set of labels wrt. R, whereas an immediate subtree has its root directly connected to the root of the tree-layer sequent.

For the following definition, we use the notation of *subscript labels* Γ_x for a set of labelled formulas Γ to refer to all formulas $x : A \in \Gamma$ that are labelled with the label x.

Definition 4.31 (From tree-layer to nesting)**.** *For a tree-layer sequent $\mathcal{R}_R, \Gamma \Rightarrow \Delta$, we define its nesting translation $\mathfrak{N}'(\mathcal{R}_R, \Gamma \Rightarrow \Delta)$ by induction on the size of $\mathcal{R}_R$.*

If $\mathcal{R}_R = \emptyset$: $\mathfrak{N}'(x : \Gamma \Rightarrow x : \Delta) = \Gamma^\bullet, \Delta^\circ$

If $\mathcal{R}_R \neq \emptyset$: $\mathfrak{N}'(\mathcal{R}_R, \Gamma \Rightarrow \Delta) = \Gamma_x^\bullet, \Delta_x^\circ, [\mathfrak{N}'(\mathfrak{S}_1)], ..., [\mathfrak{N}'(\mathfrak{S}_n)]$ with $\{\mathfrak{S}_1, ..., \mathfrak{S}_n\} = subtr(\mathcal{R}_R, \Gamma \Rightarrow \Delta)$ and $x = root(\mathcal{R}_R, \Gamma \Rightarrow \Delta)$.

To translate a fully labelled tree layered sequent, we essentially want to translate only the topmost layer while also including all necessary information from the lower layers. For this we also define the sequent $\mathfrak{S}_{max}$ which correspond roughly to the maximum layer L_{max} of $\mathfrak{S}$, as this is also where any further $\supset$R or $\Box$R rule gets applied to in the algorithm. This allows one to reduce the translation of a full tree layered sequent to definition 4.31.

Definition 4.32. *Let $\mathfrak{S}$ be a lifted sequent obtained in some derivation of the algorithm in figure 4.3.2 with its maximum layer L_{max}. We define the maximum layer sequent $\mathfrak{S}_{max}$ by the following conditions: A label x occurs in $\mathfrak{S}_{max}$ iff $x \in L_{max}$, and if $x : A^\bullet, x \leq y \in \mathfrak{S}$ and $y \in L_{max}$, then $y : A^\bullet \in \mathfrak{S}_{max}$.*

Recall for the following definition that linearly layered sequents are defined as always consisting of layers that have a rooted tree structure, just like the sequents we would get from any labIK derivation.

Definition 4.33 (From linearly layered to nesting)**.** *The nesting translation $\mathfrak{N}(\mathfrak{S})$ for a linearly layered sequent $\mathfrak{S}$ is defined as $\mathfrak{N}'(\mathfrak{S}_{max})$.*

Note that the function $\mathfrak{N}$ will only translate rules properly if the sequents in them are also fully lifted. The maximum layer might, for example, not have all the structural information that is below it due to some possible non-redundant F_1 or F_2 application. We therefore only consider sequents that we obtain from the linear proof search algorithm. One could extend the definition of translation into covering sequents that are not fully lifted (or structurally saturated) by including the notion of lifting into it. This might be interesting for a more general result that does not rely on proof search to translate bi-labelled sequents; so, as it does not contribute to our immediate goal we leave it at that.

Just like with a lot of other translations that have been established before (see e.g. [25, 33]), we would also like the translation functions to be conservative over the sequents, meaning that we can go from one formalism to the other and back while still retaining the same sequent. One such result is given by the following proposition.

Proposition 4.34. *For any nested sequent Γ, $\mathfrak{N}(\mathfrak{L}^x(\Gamma)) = \Gamma$.*

Proof. First, observe that $\mathfrak{S} = \mathfrak{L}^x(\Gamma)$ will be a sequent containing only relational atoms that form a tree on $\mathcal{R}_R$, thus a tree-layer sequent. Therefore, $\mathfrak{S}_{max} = \mathfrak{S}$. Second, see that every position of a formula inside Γ can be uniquely determined by its position in the sequent tree (multi-sets are permutation invariant), which is structurally the same as $\mathcal{R}_R$. Thus, $\mathfrak{N}(\mathfrak{L}^x(\Gamma))$ not only has the same nested form as Γ, but all formulas must occur at the same place as well, therefore proving the proposition. $\square$

The reverse direction, namely that $\mathfrak{L}^x(\mathfrak{N}(\mathfrak{S})) = \mathfrak{S}$ for a bi-labelled sequent $\mathfrak{S}$, does not hold for two reasons. Firstly, as per definition, most of the structure of $\mathfrak{S}$ actually gets lost and is unrecoverable as long as it has multiple layers. Secondly, the identity of labelled sequents relies on labels being identical; this is not guaranteed. So, the only result we would get in the other direction would be $\mathfrak{L}^x(\mathfrak{N}(\mathfrak{S})) = \mathfrak{S}_{[\overline{y}/\overline{x}]}$ for a tree-layer sequent $\mathfrak{S}$ and some $\overline{x} = (x, x_1, ..., x_n)$ and $\overline{y} = (y, y_1, ..., y_n)$ with $root(\mathfrak{S}) = y$.

The following lemma will be useful for translating general labelled sequents into nested ones, just like the context representation lemma 4.2 from the last section. This lemma allows us to consider any layer in which there are antecedent formulas and show that these formulas also occur in the topmost layer.

Lemma 4.35. *Let $\mathfrak{S}$ be a linearly layered sequent that has been obtained from the linear search algorithm after step 2 (figure 4.3.2). Let $\mathfrak{S}' \subseteq \mathfrak{S}$ be a tree-layer sequent with $\overline{x}$ a tuple of labels occurring in $\mathfrak{S}'$. Then, $\mathfrak{S}'^{\downarrow}_{[\overline{y}/\overline{x}]} \subseteq \mathfrak{S}_{max}$ for some tuple of labels $\overline{y}$ with $\overline{x} \leq \overline{y} \in \mathfrak{S}$.*

Proof. If $\mathfrak{S}'$ is in the maximum layer of $\mathfrak{S}$, it is trivial with $\overline{y} = \overline{x}$.

If $\mathfrak{S}'$ is not in the maximum layer, it must be strictly contained in some other layer L_k as it only contains R relational formulas.

Claim: We can always find a sequent $\mathfrak{S}'^{\downarrow}_{[\overline{z}/\overline{x}]} \subseteq \mathfrak{S}$ with some labels $\overline{z}$, s.t. it occurs in L_{k+1}, the layer above L_k.

To prove the claim, consider that $L_k \leq_{\mathfrak{S}} L_{k+1}$, so there are labels $x \in L_k$ and $z \in L_{k+1}$ s.t. $x \leq z \in \mathfrak{S}$. For any $u \in \overline{x}$: $u\mathcal{R}_R^{\leftrightarrow}x$, so by the lifting step and $x \leq z$, there must be a set of labels $\overline{z}$ in layer L_{k+1} s.t. $\overline{x} \leq \overline{z}$. Also, the R-relations between the labels of $\overline{x}$ are isomorphic to the ones in $\overline{z}$, as each label in $\overline{x}$ will have exactly one label in $\overline{z}$ and the R structure being preserves (condition 3 of lifting).

We can repeat this process until we reach the maximum layer. Via transitivity (condition 2 of lifting), we have $\overline{x} \leq \overline{y}$ with $\overline{x}$ and $\overline{z}$ being structurally equivalent. For computing the maximum layer sequent $\mathfrak{S}_{max}$, for all succedent formulas $u : A^{\bullet}$ with $u \in \overline{x}$ and $u \leq v$ with $v \in \overline{y}$ there is $v : A^{\bullet} \in \mathfrak{S}_{max}$. $\square$

In the following, we will show that translating each rule from labIK gives us the corresponding rule in m-NIK, although with some small caveat: The initial sequent obtained from the rule Ax might be translated into a nested sequent that is not an initial sequent. For example $x \leq y, y \leq y, y \leq z, y : p, z : p \Rightarrow y : p, z : q$ is an initial sequent in labIK, but its translation would be simply $p^{\bullet}, p^{\bullet}, q^{\circ}$, as $y \notin L_{max}$. Practically, this is not a problem, as the sequent was already initial before y got introduced, thus the algorithm we presented in figure 4.3.2 terminates before z gets introduced. This gives us for example the sequent $x \leq y, y \leq y, y : p \Rightarrow y : p, y : p \supset q$, which translates via $\mathfrak{N}$ to $p^{\bullet}, p^{\circ}, p \supset q^{\circ}$, a conclusion of Ax.

Lemma 4.36. *The translated rule*

$$Ax\ \frac{}{\mathfrak{N}(\mathcal{R}, x \leq y, x : p, \Gamma \Rightarrow \Delta, y : p)}$$

with $\mathcal{R}, x \leq y, x : p, \Gamma \Rightarrow \Delta, y : p$ linearly layered, is derivable in m-NIK under the condition that $y \in L_{max}$.

Proof. As the first step for the translation, we fill the sequent to gain $\mathcal{R}, \mathcal{R}', x \leq y, x : p, \Gamma \Rightarrow \Delta, y : p$, where $\mathcal{R}'$ contains formulas obtained from the rules Ref, $Trans$, F_1 and F_2. Compute the maximum layer sequent: By $x \leq y, x : p^\bullet$ and $y \in L_{max}$ the maximum layer sequent contains $y : p^\bullet$ and $y : p^\circ$. The translated subsequent $\mathfrak{N}(y : p \Rightarrow y : p) = p^\bullet, p^\circ$ has to occur somewhere in $\mathfrak{N}(\mathcal{R}, x \leq y, x : p, \Gamma \Rightarrow \Delta, y : p)$. Thus, $\mathfrak{N}(\mathcal{R}, x \leq y, x : p, \Gamma \Rightarrow \Delta, y : p) = \Sigma\{p^\bullet, p^\circ\}$ which is an initial sequent in m-NIK. □

Lemma 4.37. *The translated rule*

$$\bot \ \frac{}{\mathfrak{N}(\mathcal{R}, x : \bot, \Gamma \Rightarrow \Delta)}$$

with $\mathcal{R}, x : \bot, \Gamma \Rightarrow \Delta$ linearly layered, is derivable in m-NIK.

Proof. Call $\mathfrak{S} = \mathcal{R}, x : \bot, \Gamma \Rightarrow \Delta$. By lemma 4.35 we can find some formula $x' : \bot \in \mathfrak{S}^{max}$. Therefore, we can write $\mathfrak{N}(\mathfrak{S}) = \mathfrak{N}'(\mathfrak{S}_{max}) = \Sigma\{\bot^\bullet\}$ for some nested sequent $\Sigma\{\} = \mathfrak{N}(\mathcal{R}, \Gamma \Rightarrow \Delta)$. The nested sequent $\Sigma\{\bot^\bullet\}$ can be obtained immediately in m-NIK by the rule $\bot$. □

After having translated the initial sequents of labIK, for which Ax might only yields an initial sequent if it gets eagerly applied, we will also translate all the other rules like they occur in the algorithm. Let us first consider only the rules that are part of the saturation, before we translate the two $lifting$ rules.

Lemma 4.38. *For any rule application occurring in a saturation tree*

$$\frac{\mathfrak{S}_1, ..., \mathfrak{S}_n}{\mathfrak{S}}$$

with $n = 1, 2$ and $\mathfrak{S}_1, ..., \mathfrak{S}_n, \mathfrak{S}$ all being linearly layered, the translated rule

$$\frac{\mathfrak{N}(\mathfrak{S}_1), ..., \mathfrak{N}(\mathfrak{S}_n)}{\mathfrak{N}(\mathfrak{S})}$$

is derivable in m-NIK in one step.

Proof. $\wedge$L: Consider some sequent $\mathfrak{S} = \mathcal{R}, x : A, x : B, \Sigma \Rightarrow \Pi$ By lemma 4.35 there is some label x' occurring in $\mathfrak{S}$ s.t. $x \leq x' \in \mathfrak{S}$ and $x' \in L_{max}(\mathfrak{S})$. So, the formulas $x' : A^\bullet$ and $x' : B^\bullet$ are in $\mathfrak{S}_{max}$. Therefore, the formulas $A^\bullet, B^\bullet$ must occur somewhere together in the nested sequent $\mathfrak{N}(\mathfrak{S})$, letting us express this as $\mathfrak{N}(\mathfrak{S}) = \Gamma\{A^\bullet, B^\bullet\}$ for some nested sequent $\Gamma\{\} = \mathfrak{N}(\mathcal{R}, \Sigma \Rightarrow \Pi)$. In the same way, we can translate the sequent $\mathcal{R}, x : A \wedge B, \Sigma \Rightarrow \Pi$ into $\Gamma\{A \wedge B^\bullet\}$. We can connect these translated sequents through the following derivation in m-NIK.

$$\frac{\Gamma\{A^\bullet, B^\bullet\}}{\Gamma\{A \wedge B^\bullet\}} \wedge^\bullet$$

$\wedge$R: Let $\mathfrak{S} = \mathcal{R}, \Sigma \Rightarrow \Pi, x : A \wedge B$ be the conclusion of the rule application. If $x \notin L_{max}(\mathfrak{S})$, both the premise and the conclusion of the translated rule would become equal, as only succedent formulas in the maximum layer will be translated. Thus, it would be a zero-step derivation. If $x \in L_{max}$, we also have $x : A \wedge B^\circ \in \mathfrak{S}_{max}$, and therefore $\mathfrak{N}(\mathfrak{S}) = \mathfrak{N}'(\mathfrak{S}_{max}) = \Gamma\{A \wedge B^\circ\}$ with $\Gamma\{\} = \mathfrak{N}(\mathcal{R}, \Sigma \Rightarrow \Pi)$. For the same reason, we get $\mathfrak{N}(\mathcal{R}, \Sigma \Rightarrow \Pi, x : A) = \Gamma\{A^\circ\}$ and $\mathfrak{N}(\mathcal{R}, \Sigma \Rightarrow \Pi, x : B) = \Gamma\{B^\circ\}$. This translates the rule to the following m-NIK derivation.

$$\frac{\Gamma\{A^\circ\} \quad \Gamma\{B^\circ\}}{\Gamma\{A \wedge B^\circ\}} \wedge^\circ$$

The cases for $\vee$L and $\vee$R work in the same manner as the previous two.

$\supset$L: Consider the sequent $\mathfrak{S} = \mathcal{R}, x \leq y, x : A \supset B, \Sigma \Rightarrow \Pi$. Like before, we can use lemma 4.35 to find that there is some label $x' \in L_{max}(\mathfrak{S})$ with $x \leq x'$ and $x' : A \supset B^\bullet \in \mathfrak{S}_{max}$. Let us therefore write the translation as $\mathfrak{N}(\mathfrak{S}) = \mathfrak{N}'(\mathfrak{S}_{max}) = \Gamma\{A \supset B^\bullet\}$ with $\Gamma\{\} = \mathfrak{N}(\mathcal{R}, x \leq y, \Sigma \Rightarrow \Pi)$. Assume at first that $y \neq x'$. Then, $\mathfrak{N}(\mathcal{R}, x \leq y, x : A \supset B, \Sigma \Rightarrow \Pi, y : A) = \mathfrak{N}(\mathcal{R}, x \leq y, x : A \supset B, \Sigma \Rightarrow \Pi) = \Gamma\{A \supset B^\bullet\}$, as succedent formulas only are being translated if they occur in the maximum layer, which y cannot be, as it would otherwise constitute a branching of $\leq$. This gives us a zero-step derivation (we can simply ignore the other premise). Assume for the other case that $y = x'$. Now, both premises translate to $\mathfrak{N}(\mathcal{R}, x \leq y, x : A \supset B, \Sigma \Rightarrow \Pi, y : A) = \Gamma\{A \supset B^\bullet, A^\circ\}$ and $\mathfrak{N}(\mathcal{R}, x \leq y, x : A \supset B, y : A, \Sigma \Rightarrow \Pi) = \Gamma\{A \supset B^\bullet, B^\bullet\}$. The derivation connecting the translated premises to the conclusion is the rule $\supset^\bullet$.

$$\frac{\Gamma\{A \supset B^\bullet, A^\circ\} \quad \Gamma\{A \supset B^\bullet, B^\bullet\}}{\Gamma\{A \supset B^\bullet\}} \supset^\bullet$$

$\Diamond$L: Let $\mathfrak{S} = \mathcal{R}, x : \Diamond A, \Sigma \Rightarrow \Pi$ be the conclusion of some $\Diamond$L application with $x : \Diamond A$ as principal formula. By lemma 4.35, there is some formula $x' : \Diamond A^\bullet \in \mathfrak{S}_{max}$, allowing us to write the translation as $\mathfrak{N}(\mathfrak{S}) = \mathfrak{N}'(\mathfrak{S}_{max}) = \Gamma\{\Diamond A^\bullet\}$ with $\Gamma\{\} = \mathfrak{N}(\mathcal{R}, \Sigma \Rightarrow \Pi)$. Considering the linearly layered sequent $\mathfrak{S}_1 = \mathcal{R}, xRy, y : A, \Sigma \Rightarrow \Pi$ as the premise of the rule, the sequent $xRy, y : A \Rightarrow$ gets translated to the nested sequent $[A]$. As y is fresh, we do not have to consider any other formulas labelled with y that might occur inside the nesting. By lemma 4.35 and the same argument: $\mathfrak{N}(\mathfrak{S}_1) = \Gamma\{[A]\}$. The translated rule becomes the rule $\Diamond^\bullet$ in m-NIK.

$$\frac{\Gamma\{[A]\}}{\Gamma\{\Diamond A\}} \Diamond^\bullet$$

$\Diamond$R: We consider $\mathfrak{S} = \mathcal{R}, xRy, \Sigma \Rightarrow \Pi, x : \Diamond A$ with $x \in L_{max}(\mathfrak{S})$, as the derivation becomes trivial in zero steps. So, $xRy, x : \Diamond A^\circ \in \mathfrak{S}_{max} \subseteq \mathfrak{S}_{max}$. We might write the translation as $\mathfrak{N}(\mathfrak{S}) = \mathfrak{N}'(\mathfrak{S}_{max}) = \Gamma'\{\Diamond A^\circ\}$ with $\Gamma'\{\} = \mathfrak{N}(\mathcal{R}, xRy, \Sigma \Rightarrow \Pi)$. But as we also want to translate the relational formula xRy, it is better suited to also include the subtree sequent $\mathfrak{S}_y$ with $root(\mathfrak{S}_y) = y$ explicitly in the translation. We therefore get $\mathfrak{N}(\mathfrak{S}) = \Gamma\{\Diamond A^\circ, [\mathfrak{N}(\mathfrak{S}_y)]\}$ with $\Gamma\{\} = \mathfrak{N}(\mathfrak{S}_-)$ and $\mathfrak{S}_- \otimes \mathfrak{S}_y = \mathcal{R}, xRy, \Sigma \Rightarrow \Pi$. Translating $\mathfrak{S}_1 = \mathcal{R}, xRy, \Sigma \Rightarrow \Pi, x : \Diamond A, y : A$ in the same way will give us $\mathfrak{N}(\mathfrak{S}_1) = \Gamma\{\Diamond A^\circ, [A^\circ, \mathfrak{N}(\mathfrak{S}_y)]\}$. Letting $\mathfrak{N}(\mathfrak{S}_y) = \Delta$, allows us to write the translated rule as follows.

$$\frac{\Gamma\{\Diamond A^\circ, [A^\circ, \Delta]\}}{\Gamma\{\Diamond A^\circ, [\Delta]\}} \Diamond^\circ$$

The translations of $\Box$L can be obtained in a similar manner as the cases for $\Diamond$R and $\supset$L. $\Box$

In the following we consider the rule $lifting_{x:F}$ as a single rule application, as one cannot properly translate sequents that are not fully lifted. As discussed before, we could define the translation such that we might also translatesequents that are not fully structurally saturated (non-redundant applications of $Trans$, F_1 and F_2). This would translate the relational rules (Ref, $Trans$, F_1 and F_2) into simple repetition rules. The same could also be done, for example, for the saturation over $\supset$L; this rule gets only translated to $\supset^\bullet$ if $y \in L_{max}$ for the principal formula $x \leq y$. This would be guaranteed by applying $\supset$L to a formula $A \supset B$ as long as it is not redundant.

Lemma 4.39. *Consider a saturated linearly layered sequent $\mathfrak{S}$ occurring in the linear algorithm (at step 3, figure 4.3.2) with $x : F^\circ \in \mathfrak{S}$, $F = A \supset B$ or $F = \Box A$, and $x \in L_{max}(\mathfrak{S})$, the translated lifting rule*

$$\frac{\mathfrak{N}(\mathfrak{S} \otimes \mathfrak{S}{\uparrow}^{x:F})}{\mathfrak{N}(\mathfrak{S})}$$

is derivable in m-NIK in a single step.

Proof. Consider $\Box$R as the example case. Let $\mathfrak{S} = \mathcal{R}, \Gamma \Rightarrow \Delta, x : \Box A$ a linearly layered sequent with $x \in L_{max}(\mathfrak{S})$. Then, $\mathfrak{S}' = \mathcal{R}, x \leq y, yRz, \Gamma \Rightarrow \Delta, z : A$ is also linearly layered. By lifting this sequent we essentially create a copy of $L_{max}(\mathfrak{S})$ in the new maximum layer $L_{max}(\mathfrak{S}')$; we can consider only the most recent maximum layer as it must also have been lifted or it is the only layer. Let us write the translation of the conclusion as $\mathfrak{N}(\mathfrak{S}) = \Gamma\{\Box A^\circ\}$ as $x \in L_{max}(\mathfrak{S})$ holds. Then, see that the only succedent formula in the maximum layer of $\mathfrak{S}'$ is $z : A$, so we loose all other succedents, giving us $\mathfrak{N}(\mathfrak{S}') = \Gamma^{\downarrow}\{[A^\circ]\}$. Observe that the nesting is due to the fresh z, which was not due to the lifting of $L_{max}(\mathfrak{S})$ into $L_{max}(\mathfrak{S}')$. Together, $\Gamma\{\Box A^\bullet\}$ and $\Gamma^{\downarrow}\{[A^\circ]\}$ form the conclusion and premise of the m-NIK rule $\Box^\circ$.

The case for $\supset$R works similarly. ☐

Theorem 4.40 (From labIK to m-NIK)**.** *For any formula $A \in \mathcal{L}^{\Box\Diamond}$, if* labIK $\vdash \Rightarrow x : A$ *then a derivation tree for* m-NIK $\vdash A^\circ$ *can be effectively obtained.*

Proof. At first, we can construct a derivation tree of $\Rightarrow x : A$ in labIK by the algorithm in figure 4.3.2, which will exist and will be valid by proposition 4.27. We further know that all sequents will be linearly layered by construction of the derivation. Thus, we are able to translate all sequents in the lifting tree (step 1) and after the lifting (step 3a) via $\mathfrak{N}$, giving us a valid derivation in m-NIK via lemmata 4.36, 4.37, 4.38 and 4.39. ☐

It is notable that this theorem is weaker than its counterpart, theorem 4.12, as we are not able to translate proofs of full sequents. The theorem could be extended towards translating labelled tree-layer sequents as these are the labelled sequents that can be translated fully back and forth between $\mathfrak{N}$ and $\mathfrak{L}^x$.

Both translation results together (theorems 4.12 and 4.40) also yield a new way of establishing soundness and completeness. Not only do they give a different route of showing completeness wrt. the logic IK, but they also show in a direct way that labIK and m-NIK are sound and complete wrt. each other.

Corollary 4.41. *For any formula $A \in \mathcal{L}^{\Box\Diamond}$:* m-NIK $\vdash A$ *iff* labIK $\vdash A$*. Also, for any nested sequent Γ:* m-NIK $\vdash \Gamma$ *iff* labIK $\vdash \mathfrak{L}^x(\Gamma)$.

Proof. Follows immediately from theorems 4.12 and 4.40, as well as proposition 4.34. ☐

Another important corollary we can obtain from our result is that we can connect the purely propositional parts of labIK and m-NIK, which are labG3I and m-G3i (see section 3.2).

Corollary 4.42. *For any formula $A \in \mathcal{L}$: If* labG3I $\vdash A$ *then a derivation tree of* m-G3i $\vdash A$ *can be effectively obtained. If* m-G3i $\vdash A$ *then a derivation tree of* labG3I $\vdash A$ *can be effectively obtained.*

Proof. Use theorems 4.12 and 4.40 and restrict all derivations to modal-free formulas $A \in \mathcal{L}$. $\square$

A similar connection between purely classical modal calculi is not possible, as it would change the rules of the system and not simply remove some. Such a result is essentially obtained in [25].

Finally, we want to point out the fact that these translations connect a fully invertible proof system (labIK) to a system where backtracking might be needed (m-NIK). In light of how we defined the proof search algorithm for linearly layered sequents (figure 4.3.2), we can say that what corresponds to backtracking in m-NIK is nothing more than having and keeping formulas together with some extra information (i.e. their labels and $\leq$ formulas). Such structural approaches to omitting backtracking, especially in intuitionistic settings and proof search, have been around already in different forms (see e.g. [37]). A similar comparison between nested and pure sequent calculi has also been discussed recently in [43].

Chapter 5

Conclusion and Future Work

Conclusion

In this thesis, we introduced effective translations between two proof systems for the logic IK. In chapter 2, we first introduced the intuitionistic modal logic IK by its Hilbert axiomatisation and its relational Kripke semantics and saw why it can be considered a good intuitionistic modal logic. Then, in chapter 3, we looked at its proof theory, starting with different systems for intuitionistic propositional logic and then adding modalities to gain our main calculi of interest. These are a semantic bi-labelled calculus and a Maehara-style simple nested system, which we called labIK and m-NIK respectively. In the main part of the thesis, chapter 4, we have constructed effective translations between these systems. The first translation, from the simple nested formalism to the bi-labelled one, could be obtained in a relatively simple manner. By interpreting the tree structure of nestings as R-relational trees, we could simply obtain a well behaved labelled sequent on which one can emulate the m-NIK rules by using (sometimes multiple) rule applications of labIK. The reverse translation relied on a sophisticated search algorithm. This algorithm combines relational rules with the $\supset$R and $\Box$R rules to gain a new rule, *lifting*, which can be properly translated into the corresponding nested rules. Another hurdle the search algorithm helped to overcome was translating invertible rules into non-invertible ones by restricting the derivation to necessary applications of $\supset$R and $\Box$R. This also brings about a way to do proof search for m-NIK. Both translations establish a new kind of completeness between labIK and m-NIK, as well as letting us see how different these systems behave including the invertibility of rules.

Future Works

The next steps to take from here would be to incorporate extensions of IK; not only might one cover the extensions of the IS5 cube, but one could also translate proofs for extensions with Horn-Scott-Lemmon axioms, much like in the style of Lyon [32]. We expect this result to be easily obtainable by implementing propagation rules in labIK, and thus obtaining a new set of extension rules for m-NIK that is in virtue of the translation sound and complete wrt. to the logical extension.

Other variants of intuitionistic modal logic might be considered as well. This includes the so-called logic FK which can be defined semantically by leaving out the backward confluence condition in the definition of their Kripke models (see [2]). The only currently existing proof system for FK is a bi-nested system, which corresponds well to our bi-labelled labIK without the rule F_1. From this one could be able to obtain a new single nested system for FK by simply using the translation $\mathfrak{N}$. This can also be done by restricting the *lifting* rule s.t. it does not contain the backward confluence rule F_1, and translating this rule to gain a proper rule for a new system which one might call m-NFK. By construction and via the translation it would be immediate to get soundness and completeness for this new system.

Another interesting direction to investigate is considering other systems for IK, namely the Simpson calculus and its nested notational variant (see [32]) and connect, for example, the single succedent nested calculus with m-NIK. While translating from the single succedent system, with more invertible rules, into the multi succedent one is trivial due to weakening, the reverse might be obtainable in a similar manner as the translation between G3i and m-G3i in [34].

Besides these more theoretical investigations, it could be interesting to implement these results into a theorem prover. Not only might one want to implement the search algorithms presented here, or a variant in the style of [23, 24] that can also find counter models, it could also be fruitful to look at implementing the translations themselves.

Bibliography

[1] Amon Avron. "The Method of Hypersequents in the Proof Theory of Propositional Non-classical Logics". In: *Logic: from Foundations to Applications: European logic colloquium.* Oxford University Press, 1996.

[2] Philippe Balbiani, Han Gao, Çiğdem Gencer, and Nicola Olivetti. "A Natural Intuitionistic Modal Logic: Axiomatization and Bi-Nested Calculus". In: *32nd EACSL Annual Conference on Computer Science Logic (CSL 2024)*. Ed. by Aniello Murano and Alexandra Silva. Vol. 288. Leibniz International Proceedings in Informatics (LIPIcs). Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024, 13:1–13:21. DOI: 10.4230/LIPIcs.CSL.2024.13.

[3] Gianluigi Bellin, Valeria De Paiva, and Eike Ritter. "Extended Curry-Howard Correspondence for a Basic Constructive Modal Logic". In: (May 2003).

[4] Nuel D. Belnap. "Display Logic". In: *Journal of Philosophical Logic* 11.4 (1982), pp. 375–417.

[5] Evert Willem Beth. *The Foundations of Mathematics a Study in the Philosophy of Science.* Amsterdam, Netherlands: Harper & Row, 1959.

[6] Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic.* Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2001.

[7] Milan Božić and Kosta Došen. "Models for Normal Intuitionistic Modal Logics". In: *Studia Logica* 43.3 (1984), pp. 217–245. DOI: 10.1007/bf02429840.

[8] Kai Brünnler. "Deep Sequent Systems for Modal Logic". In: *Archive for Mathematical Logic* 48.6 (2009), pp. 551–577. DOI: 10.1007/s00153-009-0137-3.

[9] Robert A. Bull. "Cut Elimination for Propositional Dynamic Logic Without". In: *Zeitschrift fur mathematische Logik und Grundlagen der Mathematik* 38.1 (1992), pp. 85–100. DOI: 10.1002/malq.19920380107.

[10] Haskell B. Curry. *Foundations of Mathematical Logic.* McGraw-Hill series in higher mathematics. McGraw-Hill, 1963.

[11] Tiziano Dalmonte, Björn Lellmann, Nicola Olivetti, and Elaine Pimentel. "Hypersequent calculi for non-normal modal and deontic logics: countermodels and optimal complexity". In: *Journal of Logic and Computation* 31.1 (Dec. 2020), pp. 67–111. DOI: 10.1093/logcom/exaa072.

[12] Anupam Das and Sonia Marin. "On Intuitionistic Diamonds (and Lack Thereof)". In: *Automated Reasoning with Analytic Tableaux and Related Methods.* Ed. by Revantha Ramanayake and Josef Urban. Springer Nature Switzerland, 2023, pp. 283–301.

[13] Albert G. Dragalin. *Mathematical Intuitionism: Introduction to Proof Theory*. Translations of mathematical monographs. American Mathematical Society, 1988.

[14] Roy Dyckhoff. "Intuitionistic Decision Procedures Since Gentzen". In: *Advances in Proof Theory*. Ed. by Reinhard Kahle, Thomas Strahm, and Thomas Studer. Springer International Publishing, 2016, pp. 245–267. DOI: 10.1007/978-3-319-29198-7_6.

[15] Roy Dyckhoff and Sara Negri. "Proof Analysis in Intermediate Logics". In: *Arch. Math. Log.* 51 (Feb. 2012), pp. 71–92. DOI: 10.1007/s00153-011-0254-7.

[16] W. B. Ewald. "Intuitionistic Tense and Modal Logic". In: *The Journal of Symbolic Logic* 51.1 (1986), pp. 166–179.

[17] Gisèle Fischer Servi. "Axiomatisations for some intuitionistic modal logics". In: *Rendiconti del Seminario Matematico - PoliTO* 42.3 (1984), pp. 179–194.

[18] Gisèle Fischer Servi. "On Modal Logic with an Intuitionistic Base". In: *Studia Logica: An International Journal for Symbolic Logic* 36.3 (1977), pp. 141–149.

[19] Frederic B. Fitch. "Intuitionistic modal logic with quantifiers". In: *Portugaliae mathematica* 7.2 (1948), pp. 113–118.

[20] Melvin Fitting. *Intuitionistic Logic Model Theory and Forcing*. 1969.

[21] Melvin Fitting. "Nested Sequents for Intuitionistic Logics". In: *Notre Dame Journal of Formal Logic* 1 (2014). DOI: 10.1215/00294527-2377869.

[22] Gerhard Gentzen. "Untersuchungen über das logische Schließen. I". In: *Mathematische Zeitschrift* 39 (1935), pp. 176–210.

[23] Marianna Girlando, Roman Kuznets, Sonia Marin, Marianela Morales, and Lutz Straßburger. "A Simple Loopcheck for Intuitionistic K". preprint. 2024.

[24] Marianna Girlando, Roman Kuznets, Sonia Marin, Marianela Morales, and Lutz Straßburger. "Intuitionistic S4 is decidable". In: *2023 38th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*. IEEE, 2023. DOI: 10.1109/lics56636.2023.10175684.

[25] Rajeev Goré and Revantha Ramanayake. "Labelled tree sequents, tree hypersequents and nested (Deep) sequents". In: *Advances in Modal Logic* 9 (Jan. 2014).

[26] Stig Kanger. *Provability in Logic*. Stockholm, Almqvist & Wiksell, 1957.

[27] Ryo Kashima. "Cut-Free Sequent Calculi for Some Tense Logics". In: *Studia Logica: An International Journal for Symbolic Logic* 53.1 (1994), pp. 119–135.

[28] Oiva Ketonen. "Untersuchungen Zum Prädikatenkalkül". In: *Journal of Symbolic Logic* 10.4 (1945), pp. 127–130.

[29] Stephen Cole Kleene. *Introduction to Metamathematics*. Groningen: P. Noordhoff N.V., 1952.

[30] Roman Kontchakov, Agi Kurucz, and Michael Zakharyaschev. "Undecidability of First-Order Intuitionistic and Modal Logics with Two Variables". In: *The Bulletin of Symbolic Logic* 11.3 (2005), pp. 428–438.

[31] Roman Kuznets and Lutz Straßburger. "Maehara-style modal nested calculi". In: *Archive for Mathematical Logic* 58 (May 2019), pp. 359–385. DOI: 10.1007/s00153-018-0636-1.

[32] Tim S. Lyon. "Nested Sequents for Intuitionistic Modal Logics via Structural Refinement". In: *Automated Reasoning with Analytic Tableaux and Related Methods*. Springer International Publishing, 2021, pp. 409–427. DOI: 10.1007/978-3-030-86059-2_24.

[33] Tim S. Lyon. “On Deriving Nested Calculi for Intuitionistic Logics from Semantic Systems”. In: *Lecture Notes in Computer Science*. Springer International Publishing, Dec. 2019, pp. 177–194. DOI: 10.1007/978-3-030-36755-8_12.

[34] Shôji Maehara. “Eine Darstellung der Intuitionistischen Logik in der Klassischen”. In: *Nagoya Mathematical Journal* 7 (1954), pp. 45–64. DOI: 10.1017/S0027763000018055.

[35] Sonia Marin. “Modal proof theory through a focused telescope”. Theses. Université Paris Saclay, 2018.

[36] Sonia Marin, Marianela Morales, and Lutz Straßburger. “A fully labelled proof system for intuitionistic modal logics”. In: *Journal of Logic and Computation* 31.3 (2021), pp. 998–1022. DOI: 10.1093/logcom/exab020.

[37] Grigori Mints. “Proof-search”. In: *A Short Introduction to Intuitionistic Logic*. Springer US, 2000, pp. 75–81. DOI: 10.1007/0-306-46975-8_11.

[38] Sara Negri. “Kripke completeness revisited”. In: *Acts of knowledge*. College publications, 2009, pp. 233–264.

[39] Sara Negri. “Proof Analysis in Modal Logic”. In: *Journal of Philosophical Logic* 34 (2005), pp. 507–544. DOI: 10.1007/s10992-005-2267-3.

[40] Sara Negri. “Proofs and Countermodels in Non-Classical Logics”. In: *Logica Universalis* 8 (2014), pp. 25–60. DOI: 10.1007/s11787-014-0097-1.

[41] Hiroakira Ono. “On Some Intuitionistic Modal Logics”. In: *Publications of the Research Institute for Mathematical Sciences* 13 (Dec. 1977). DOI: 10.2977/prims/1195189604.

[42] Katudi Ono. “Logische Untersuchungen Über Die Grundlagen der Mathematik”. In: *Journal of Symbolic Logic* 4.2 (1939), pp. 89–90.

[43] Elaine Pimentel, Revantha Ramanayake, and Björn Lellmann. “Sequentialising Nested Systems”. In: *Automated Reasoning with Analytic Tableaux and Related Methods*. Ed. by Serenella Cerrito and Andrei Popescu. 28th International Conference, TABLEAUX 2019 ; Conference date: 03-09-2019 Through 05-09-2019. Springer, 2019, pp. 147–165. DOI: 10.1007/978-3-030-29026-9_9.

[44] Jan von Plato. *Saved from the Cellar. Gerhard Gentzen’s Shorthand Notes on Logic and Foundations of Mathematics*. 1st ed. Sources and Studies in the History of Mathematics and Physical Sciences. Springer Cham, 2017.

[45] Gordon Plotkin and Colin Stirling. “A framework for intuitionistic modal logics: extended abstract”. In: *Proceedings of the 1986 Conference on Theoretical Aspects of Reasoning about Knowledge*. TARK ’86. Morgan Kaufmann Publishers Inc., 1986, pp. 399–406.

[46] Francesca Poggiolesi. “The Tree-hypersequent Method for Modal Propositional Logic”. In: *Trends in Logic: Towards a Mathematical Philosophy*. Ed. by D. Makinson, J. Malinowski, and H. Wansing. Springer, 2009, pp. 31–51.

[47] Dag Prawitz. *Natural Deduction: A Proof-Theoretical Study*. Dover Publications, 1965.

[48] Arthur N. Prior. *Time and Modality*. Westport, Conn.: Greenwood Press, 1955.

[49] Kurt Schütte. “Schlußweisen-Kalküle der Prädikatenlogik”. In: *Mathematische Annalen* 122 (1950), pp. 47–65.

[50] Alex K. Simpson. “The Proof Theory and Semantics of Intuitionistic Modal Logic”. PhD thesis. University of Edinburgh, 1994.

[51] Vladimir Sotirov. “Modal Theories with intuitionistic logic”. In: *Proceedings of the Conference on Mathematical Logic, Sofia 1980*. Bulgarian Academy of Sciences, 1984, pp. 139–171.

[52] Lutz Straßburger. “Cut Elimination in Nested Sequents for Intuitionistic Modal Logics”. In: *Foundations of Software Science and Computation Structures*. Ed. by Frank Pfenning. Springer Berlin Heidelberg, 2013, pp. 209–224.

[53] Anne S. Troelstra and Helmut Schwichtenberg. *Basic Proof Theory*. 2nd ed. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2000.

[54] Luca Vigano. *Labelled Non-Classical Logics*. Boston: Kluwer Academic Publishers, 2000.

[55] Heinrich Wansing. “Sequent Systems for Modal Logics”. In: 8 (Jan. 2002). DOI: 10.1007/978-94-010-0387-2_2.

[56] Duminda Wijesekera. “Constructive modal logics I”. In: *Annals of Pure and Applied Logic* 50.3 (1990), pp. 271–301. DOI: 10.1016/0168-0072(90)90059-B.