Unusual proof systems for modal logics with applications to decision problems

Marianela Morales Elena

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2023IPPAX129

Thèse de doctorat

ÉCOLE POLYTECHNIQUE

IP PARIS

Unusual proof systems for modal logics with applications to decision problems

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à École polytechnique

École doctorale n°626 École doctorale de l'Institut Polytechnique de Paris (EDIPP)
Spécialité de doctorat : Informatique

Thèse présentée et soutenue à Palaiseau, le 8 décembre 2023, par

MME MARIANELA MORALES

Composition du Jury :

M. David Baelde Professeur, ENS Rennes	Président
M. Carlos Areces Professeur, Universidad Nacional de Córdoba	Rapporteur
Mme. Elaine Pimentel Professeur Associé, University College London	Rapporteur
Mme. Delia Kesner Professeur, Université Paris Cité	Examinateur
M. Lutz Straßburger Directeur de Recherche, Inria Saclay & LIX	Directeur de thèse

Como van a convencerme de que la magia no existe

Acknowledgements

A few years ago, I embarked on my PhD journey, and I would like to draw inspiration from my favorite TV show to describe what these years have meant to me:

It was at times a long, difficult road. But I'm glad it was long and difficult, because if I hadn't gone through hell to get there, the lesson might not have been as clear.

- How I met your mother -

Truly it has been a road filled with challenges, but I could not have make it without the unwavering support of many people who stood beside me.

First, I would like to express my gratitude to my supervisor Lutz Straßburger. Thanks for all the time he dedicated to this thesis and for the support he gave me (specially when I was panicking). Thanks for his mentorship and for introducing me to a lot of interesting (and nice!) people.

I would like to strongly thank Sonia Marin and Marianna Girlando, for made the academic journey more enjoyable. Thanks for their valuable feedback, their stimulating discussions about proof theory (and ice-creams and pineapple pizzas!) and for being a great inspiration to me. Thanks to them and to Roman Kuznets for collaborating in results of this thesis.

I would also like to extend my gratitude to Carlos Areces, the person whose passion for research inspired me to embark on this doctoral studies in modal logics. Thanks to be part of this thesis jury with Elaine Pimentel, Delia Kesner and David Baelde.

I am grateful to the École Polytechnique and to INRIA Saclay to give me the opportunity to start my academic career. Thanks to my PARTOUT team for the company and support all these years.

A personal thanks to my parents and my sisters for their unwavering support and for being with me in every step of the way. Even though they are on the other side of the world, I have felt their love and companionship constantly throughout these years.

I want to express a special appreciation to the person who has patiently listened to me during my moments of frustration while working on my thesis: Francisco. Your presence has made the tough times easier and the good times even better. Thank you for always believing in me.

Last but not least, I want to thank my friends for their understanding, patience, and motivation during the ups and downs of this doctoral journey. Specially thanks to *las Beshas*, *los Gatos papa*, *la Flia Cordobesa*, Flor, Roni, Marion, Nachi, Joaco and *FaMAF en exilio*. Your friendship means the world to me, and I would not have been able to do it without any of you.

Résumé

Cette thèse explore rigoureusement les logiques modales, les analysant du point de vue de la théorie de la preuve. Plus précisément, notre principal objectif se concentre sur l'étude approfondie du cadre intuitionniste des logiques modales en utilisant la théorie de la preuve. En particulier, nous travaillons avec des systèmes de séquents étiquetés pour examiner différentes variantes des logiques modales intuitionnistes et pour développer des procédures de décision afin d'établir la décidabilité de logiques spécifiques.

Ce travail débute par l'introduction des concepts fondamentaux pour le développement de notre contribution. Dans les Chapitres 2 et 3, nous présentons la syntaxe et la sémantique des logiques modales classiques et intuitionnistes, y compris les axiomatisations de Hilbert et de Gentzen. Dans les deux contextes, nous avons concentré notre attention sur la déduction étiquetée, plus précisément sur les systèmes de séquents étiquetés en raison de nos intérêts de recherche dans l'étude de l'approche intuitionniste à l'aide de ce formalisme. Dans ces chapitres introductifs, nous avons également présenté les systèmes de preuves étiquetés de Negri [49] et de Simpson [59] pour les logiques modales classiques et intuitionnistes, respectivement.

À partir du Chapitre 4, nous introduisons nos contributions aux logiques modales intuitionnistes. Les systèmes mentionnés précédemment ont constitué notre première compréhension pour la conception et le développement de notre première contribution : notre *système de séquentiel entièrement étiqueté pour les logiques modales intuitionnistes*. Ce travail conjoint avec Marin et Straßburger [43] présente un système de séquents étiquetés pour capturer les logiques modales intuitionnistes. Ce système étiqueté présente non pas un (comme celui présenté par Simpson), mais deux symboles de relation apparaissant dans les séquents : l'un pour la relation d'accessibilité R associée à la sémantique de Kripke pour les logiques modales normales et l'autre pour la relation de préordre $\leq$ associée à la sémantique de Kripke pour la logique intuitionniste. Cela place notre système en correspondance étroite avec la sémantique de Kripke birelationnelle standard pour les logiques modales intuitionnistes. Nous montrons que ce système est sonore et complet, et nous présentons une preuve de son élimination des coupes. Ce système est d'abord présenté pour la logique modale intuitionniste IK, mais ensuite, en conséquence de l'explicitation des deux relations dans le système, il peut englober une gamme plus large de logiques modales intuitionnistes : en particulier, nous présentons des extensions du système de séquents entièrement étiqueté avec des axiomes intuitionnistes unilatéraux de Scott-Lemmon et avec des axiomes de Scott-Lemmon de chemin, ce qui nous permet d'avoir un cadre étiqueté complètement général pour capturer toutes les logiques modales intuitionnistes du cube $\mathsf{S5}$. Cette contribution est présentée dans le Chapitre 4.

Le système de séquents entièrement étiqueté devient l'un des ingrédients clés pour aborder le *problème de décidabilité de diverses logiques modales intuitionnistes dans le cube* $\mathsf{S5}$.

Dans le Chapitre 5, nous commençons à travailler sur les problèmes de décision pour des logiques "plus simples", au sens où leur décidabilité était déjà connue. En particulier, nous travaillons d'abord avec des logiques où la transitivité ne fait pas partie des conditions restreignant la classe de cadres avec lesquels nous travaillions, comme c'est le cas des logiques modales intuitionnistes $\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}$ et $\mathsf{IK5}$. Nous fournissons pour elles une procédure de décision constructive qui, étant donnée une formule, produit soit une preuve montrant que la formule est valide, c'est-à-dire qu'elle est un théorème de la logique, soit un contre-modèle fini falsifiant la formule, et montrant donc que la formule n'est pas démontrable dans la logique. Cette procédure est basée sur notre système de séquents entièrement étiqueté. Ce résultat sert également d'introduction solide pour approfondir notre travail dans des logiques dont la décidabilité était inconnue jusqu'à cette année, comme c'est le cas de la logique intuitionniste $\mathsf{IS4}$ [21]. En particulier, toutes les notions présentées dans le Chapitre 5 sont développées dans le but d'être applicables à toutes les logiques modales intuitionnistes du cube $\mathsf{S5}$. De plus, l'algorithme de recherche est développé avec la flexibilité d'être personnalisé pour chaque logique modale intuitionniste spécifique que nous souhaitons utiliser, incorporant les contraintes nécessaires pour démontrer sa terminaison dans chaque cas. Nous étendons ce résultat pour étudier la décidabilité de la logique $\mathsf{IS4}$, présentant ainsi une autre contribution significative de cette thèse dans le Chapitre 6.

Le problème de décision de la logique modale intuitionniste $\mathsf{IS4}$ est un problème ouvert depuis près de trente ans depuis qu'il a été posé dans la thèse de doctorat de Simpson en 1994. Avec Girlando, Kuznets, Marin et Straßburger [21], nous prouvons que la *logique intuitionniste* $\mathsf{IS4}$ *est décidable*. Nous sommes capables d'obtenir ce résultat grâce à deux ingrédients clés. Le premier est l'utilisation du système de séquents entièrement étiqueté (comme nous le faisons pour les autres logiques mentionnées dans le Chapitre 5), car ce système hérite des avantages des systèmes étiquetés pour la logique propositionnelle intuitionniste et les logiques modales classiques : en particulier, toutes les règles d'inférence sont inversibles (c'est-à-dire que nous ne supprimons jamais d'informations de bas en haut dans la recherche de preuves) et il existe une correspondance directe entre les séquents et les modèles qui nous permet de construire un contre-modèle en interprétant (et en étendant) le séquent étiqueté auquel la recherche de preuves se termine. Et deuxièmement, en introduisant un mécanisme que nous avons appelé le *unfolding*, qui fournit une véritable preuve lorsque l'algorithme de recherche se termine dans des séquents axiomatiques. En d'autres termes, nous montrons que nous pouvons préserver la cohérence si nous organisons la recherche de preuves de manière systématique.

Toutes ces contributions se situent dans le cadre intuitionniste des logiques modales. Dans le Chapitre 7, nous présentons notre travail dans un fragment de l'approche classique. Ce chapitre est un travail en cours dans le formalisme de l'inférence profonde en utilisant la méthodologie de la *théorie de la preuve subatomique*. Cette méthode traite les atomes comme des connecteurs binaires, conduisant à une forme uniforme de toutes les règles d'inférence. Cela réduit énormément le nombre de cas dans l'analyse des coupes pour l'élimination. Cela soulève la question : *comment cette méthodologie peut-elle traiter les modalités alors qu'elles sont des connecteurs unaires?*. Nous commençons à explorer ce nouveau formalisme et nous obtenons un système de preuve subatomique pour le fragment linéaire de la logique modale classique. Nous prouvons son élimination des coupes par scindement et, en résultat de cette preuve, nous prouvons l'admissibilité des règles ascendantes qui sont les coupes.

Contents

1 Introduction **1**
- 1.1 Logic to reason about computation . . . 1
- 1.2 Proof theory for modal logics . . . 3
- 1.3 Decision problems for modal logics . . . 4
- 1.4 Structure of the thesis . . . 6

2 Classical modal logic **7**
- 2.1 Syntax and semantics . . . 7
- 2.2 Axiomatizations *à la Hilbert* . . . 9
- 2.3 Axiomatizations *à la Gentzen* . . . 12
  - 2.3.1 Labelled sequents . . . 13

3 Intuitionistic modal logic **15**
- 3.1 Syntax and semantics . . . 15
- 3.2 Axiomatizations *à la Hilbert* . . . 17
- 3.3 Labelled sequents . . . 19

4 Fully labelled proof system **22**
- 4.1 Motivation . . . 22
- 4.2 The proof system . . . 24
- 4.3 Completeness . . . 28
- 4.4 Soundness . . . 32
- 4.5 Cut Admissibility . . . 34
- 4.6 Extensions . . . 41
  - 4.6.1 One-sided Scott-Lemmon axioms . . . 41
  - 4.6.2 Path Scott-Lemmon axioms . . . 45
- 4.7 Conclusion . . . 49

5 Decision procedure for intuitionistic modal logics **50**
- 5.1 Building our search algorithm . . . 51
- 5.2 Search Algorithm . . . 56
  - 5.2.1 Semi-saturation . . . 57
  - 5.2.2 Saturation . . . 58
  - 5.2.3 Lifting saturation . . . 60
  - 5.2.4 Proof search algorithm . . . 61
- 5.3 Countermodel construction . . . 63
- 5.4 Proof construction . . . 65

5.5 Termination 66
5.6 Conclusion 67

6 Decision problem for IS4 **68**
6.1 Why is it a hard problem? 68
6.2 Search algorithm for IS4 71
6.2.1 Semi-saturation 73
6.2.2 Saturation 73
6.2.3 Lifting saturation 77
6.2.4 Loop-saturation 78
6.2.5 Proof search algorithm 82
6.3 Countermodel construction 83
6.4 Proof construction: Unfolding 87
6.5 Termination 97
6.6 Conclusion 99

7 Subatomic logic for modalities **100**
7.1 From deep inference to subatomic logic 100
7.2 Subatomic logic for classical propositional logic 103
7.3 Subatomic modal logic 105
7.3.1 The subatomic system 106
7.3.2 Splitting 108

8 Conclusion **128**
8.1 A tour of our thesis 128
8.2 Exploring next steps 130

Bibliography **131**

Chapter 1

Introduction

In this thesis our primary focus centers on the comprehensive study of *intuitionistic modal logics using proof theory*. In particular, we work with labelled sequent systems to investigate different variants of intuitionistic modal logics and to develop decision procedures for establishing the decidability of specific logics.

In the upcoming sections, we are going to outline the path we have taken to explore our main research interests. In particular, we start in Section 1.1 presenting the field of our studies, which is computer science from a mathematical perspective or as it is called *Computational Logic*. We precise our focus on studying the family of modal logics through the lens of proof theory in Section 1.2. We continue with Section 1.3 providing a comprehensive overview of related work on decision problems for modal logics. Finally, in Section 1.4 we outline the structure of this thesis.

1.1 Logic to reason about computation

Computer Science is a field which lies at the intersection of engineering and mathematics, making it difficult to classify as solely one or the other. From one side, it involves designing, constructing, and evaluating computer systems from a software development point of view, which aligns with the principles of engineering. However, from a theoretical perspective, the fundamental concepts of computer science are explained mathematically. As a result, the discipline of computer science cannot be one from the two points of view since from a practical perspective, the goal is to solve problems with computer programs; but understanding the feasibility of solving a problem and its level of difficulty requires studying computer science from a mathematical perspective.

In this thesis, we focus on computer science from a mathematical point of view. Since there are many connections between mathematics and computation, we have to clarify that our thesis is focus on *Computational Logic* [55]. Computational logic is the *use of logic to reason about computation*. By using logic, we can describe the behavior of a computing system in a precise and rigorous way, and we can reason about its correctness, efficiency, and other properties. In [27], there are examples of applications of computational logic in different areas and the impact which is generated. Let us discuss some of those examples to understand the important role of computational logic and why we are interested in studying this field.

Complexity Theory [31] is one of the applications of logic in theoretical computer science.

It investigates the question *"How much time and how much memory space is needed to solve a particular problem?"*. This measure is done by classifying a problem into complexity classes (for example: polynomial time (P), non-deterministic polynomial time (NP), polynomial space (PSPACE), exponential time (EXPTIME), etc). Natural complexity classes, as the ones mentioned before, have natural descriptive characterizations. Descriptive complexity is the field in charge of studying them. One example is Fagin's theorem [14] which characterizes the class NP purely by logic, with no mention of machines or time. It characterizes the class with the existential fragment of Second-Order Logic. Therefore, logic has proven to be a valuable tool for addressing some of the basic questions in complexity theory.

Another application of computational logic is *Databases*. Logic and databases have been intimately connected since the birth of database systems in the early 1970's. In particular, First-Order Logic (FOL) lies at the core of modern database systems, and the standard query languages such as Structured Query Language (SQL) and Query-By-Example (QBE) are syntactic variants of FOL. More powerful query languages are based on extensions of FOL with recursion. The impact of logic on databases is one of the most remarkable examples of the effectiveness of logic in computer science.

Logic plays a crucial role in establishing the fundamental properties of programming languages, making it a key tool in the software product development process. *Type theory* [54] emerged as a unifying conceptual framework for the design, analysis, and implementation of programming languages. It provides a foundation for developing logics of program behavior that are essential for reasoning about programs and it suggests new techniques for implementing compilers that improve the efficiency and integrity of generated code.

Reasoning about knowledge has been shown to play a key role in such diverse fields as *distributed computing*, *game theory*, and *artificial intelligence*. The main concern is the connection between knowledge and action. This initiates the study of epistemic logics [64], in which knowledge and interaction are represented in models of *possible worlds* (we are going to talk about possible worlds in Chapter 2), providing a formal framework to represent the kind of computational systems that we mentioned before.

We can also name *Software Verification* as another area where computation and logic are intrinsic parts of the problem: computer programs and their behaviour are formalized in some logical language, and properties are verified on this representation.

We have presented various instances of computational problems addressed through a logical methodology. These serve as illustrative examples where adopting a logical approach has directly contributed to advances in the field of computer science. In this thesis, we will investigate a family of logics that can be used to talk about relational structures, i.e. graphs, which allows their use in various domains: linguistics, game theory, software verification, etc. This logic is well-known as *Modal Logic* [5, 6]. What is called modal logic describes the behaviour of the abstract modalities $\Box$ and $\Diamond$, but covers a wide range of 'real' modalities in linguistic expressions or "modes of truth" like: necessity, possibility, belief, knowledge, obligation, and permission. It is for this reason that modal logics are particularly useful in a wide range of fields. In particular, they are such a recurring choice in computer science: in general, modal logics have a good balance between expressiveness and computational behavior.

There are many modal logics, each of them with its particular properties. In particular, the language of *classical modal logic* can be seen as an extension of propositional logic but it has already been shown that it is a fragment of first-order logic. To be more precise, it is a

fragment of first-order logic with two variables, which has been shown to be decidable [57]. Our question now is: *what about the extensions of this logic? Have they good computational behaviour? What about the intuitionistic version of modal logics? And the extensions of the intuitionistic version?* In this thesis, we are going to explore these logics through the eye of proof systems and study some of those questions. We introduce formally the modal logics we are going to work with in Chapters 2 and 3.

1.2 Proof theory for modal logics

As it was mentioned in the previous section, we focus our research in modal logics and we are interested in study them using *proof theory*. This discipline appears from the need to have a purely mathematical language to formalize the proofs and understand their properties. Frege pioneered this discipline suggesting that proofs can be considered themselves as objects of mathematical study in 1879 in his Begriffschrift [17]. Hilbert followed Frege's ideas and proposed the definition of a deductive system to formalize reasoning [29].

Another of the important figures in proof theory was Gentzen, who introduced fundamental proof-theoretic methods to prove results in mathematical logic in 1934 [19, 20]. In particular, Gentzen was the one who developed the *sequent calculus*, an alternative representation of the proofs that promote the inference rules over axioms and which provides an intuitive way of searching for a proof. The *cut-elimination* theorem or also known as *Hauptsatz* introduced by Gentzen, states that any purely logical proof in the sequent calculus can be transformed into a normal analytic form. This means that in sequent calculus, any proof can be performed without lemmas. In order to prove the Hauptsatz, one needs to show that a rule called cut is redundant in the system. In other words, it states that any proof that has a proof in the sequent calculus using cut also has a proof without this rule.

Another of Gentzen's main contributions was the parallel study of classical and intuitionist logic. Intuitionistic logics arise with Brouwer [7] as a formalization of constructive reasoning (i.e. the existence of an object is equivalent to the possibility of its construction), rejecting the *Principle of the Excluded Middle* which establishes that a proposition must be true or false. Whereas in intuitionistic logic there can be uncertainty as to whether or not a proposition holds, in classical logic the principle of the excluded middle is accepted. In this work, it is worth noting that by intuitionistic logic we mean basic propositional intuitionistic logic. Gentzen observed that, in the sequent calculus, when we define proof systems for an intuitionistic calculus, it is possible to obtain such systems simply by a syntactical restriction of the classical one.

The intuitionist approach was used on different logics, in particular, and in which we are interested in this thesis, it was used in modal logics. The foundation of modal logics started with Aristotle and with an important development in the Middle Ages. Its consolidation occurred in the late 1950's and early 1960's with the development of a semantics based on *possible worlds* introduced by Kripke [34] (hence the name of Kripke semantics) which allows us to understand modal logic as a language for graphs or as a language for describing processes, that is, to see the elements of the graph as a set of computational states and to see the relations as actions that transform one state into another.

Proof theory has been used to study various proof systems for modal logics, including sequent calculi and natural deduction systems. These proof systems provide a formal representation of the rules for manipulating modal formulas and for deriving conclusions

from them. By studying the properties of these proof systems, we can gain insight into the complexity of the corresponding modal logics and develop proof search algorithms that can efficiently find proofs in these logics, which it is one of the fields we are interested in.

For many years the study of modal logics was based on classical reasoning but the interest in intuitionistic versions come much later for two different reasons: from one perspective, logicians were motivated by a theoretical interest in obtaining intuitionistically relevant versions of modal logics, and simultaneously, specific applications in computer science naturally led to the emergence of some modal logics with a constructive flavour. In this thesis, we are interested in study both classical and intuitionistic approaches for modal logics, but we are going to focus mainly in the intuitionistic setting. In particular, we study different proof systems formalisms for classical and intuitionistic logic such as sequent calculus and deep inference. Inside the formalism of *deep inference*, we use a methodology called *subatomic proof theory* [1] to obtain a subatomic proof system for a fragment of classical modal logic (presented in Chapter 7). Then, we focus mainly on the intuitionistic approach: with Marin and Straßburger [43], we work on the design and development of a labelled sequent system which is sound and complete, and we also prove its cut-elimination (presented in Chapter 4). This labelled system allows us to study the decision problem for some intuitionistic modal logics which decidability was already known but we present here a new approach: this is for the logics $\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}$ and $\mathsf{IK5}$ (presented in Chapter 5). However, this system allows us to study decision problems for extensions of the intuitionistic setting such as intuitionistic modal logic $\mathsf{IS4}$ and $\mathsf{IK4}$ which decidability until now was unknown [21]. Let us discuss a bit more in detail about the decision problems for modal logics in the next section.

1.3 Decision problems for modal logics

We have already mentioned decidability of a logic in Section 1.1. More intuitively speaking, a decision problem for a logic is a computational problem that takes as input a formula or statement in the logic and returns either "yes" or "no", depending on whether the formula is valid or invalid in the logic. The goal is to design an algorithm or procedure that can solve the decision problem efficiently, meaning that it can determine the answer in a reasonable amount of time. To be more formal, for a logic to be *decidable*, there must be a recursive procedure which determines, for each formula, whether or not it is a theorem of the logic, concluding with either a proof in a deductive system or a suitable countermodel.

Gentzen was the first to show decidability of intuitionistic propositional logic in 1935, using the sequent calculus he had designed [19]. His approach was to bound the number of consequences inferred from some given initial sequents. It was later observed (independently by Ono [51], Ketonen [32] and Kleene [33]) that if structural rules were built in into the notation, it allowed for a root-first proof search approach. Namely, search for a sequent calculus proof until either it terminates with a proof or it reaches a sequent which already occurred along the branch leading to it, at which point it is possible to stop proof search and reconstruct a Kripke countermodel from that branch.

In the case of modal logics (on a classical propositional base), decidability has been investigated later on. Ladner defines decision procedures for some common modal logics such as $\mathsf{S4}$ based purely on Kripke semantics, without reference to sequent calculi [36]. Otherwise, the procedure described above can similarly be applied to modal logic $\mathsf{S4}$, as it also characterizes the logic of reflexive transitive Kripke frames. Generally the approach

via proof search in the sequent calculus is available for those modal logics that have a cut-free sequent system [56] (or, in certain cases, a nested sequent system using a similar strategy [8]).

Labelled sequents appear as a good choice since they internalize certain elements from Kripke semantics into the syntax of sequents, which turn out to have several interesting consequences to build decision procedures for intuitionistic propositional logic [12] as well as modal logics. First, in the setting of intuitionistic propositional logic, it is not necessary to restrict any rule to a single-conclusion. This means that all the rules can be made invertible (i.e. never deleting information in bottom-up proof search) and that the proof search procedure becomes deterministic: no need for backtracking. Then, in both settings, the loop-check for ensuring termination is not done along a branch of the proof search tree, but within a topmost sequent itself, between two labels which carry the same formulas. Finally, in the case that a loop is found, the countermodel can be build directly from the topmost sequent. Indeed, the syntax of labelled sequents makes it easy to read off a countermodel from a sequent by simply substituting labels or adding back edges, both construction being available directly in the labelled syntax.

As we mentioned in the previous section, we are interested in the study of the decision problems for logics which decidability is unknown, as it is the case of the intuitionistic S4. Intuitionistic modal logic IS4 is a way to combine intuitionistic propositional logic and modal logic S4. It is obtained as an extension of intuitionistic modal logic IK. This variant of IK and its extensions were first studied in [58, 52] and investigated in detail in [59], where decidability was shown for most logics in the intuitionistic modal S5-cube (we are going to present the S5-cube in Chapter 2 in Figure 2.2). The decidability of the logic IS5 had been shown earlier by Mints [47], but the decidability for IK4 and IS4 remained open. Indeed, the question of IS4's decidability hides a lot more complexity than either of its parents. As it can be interpreted on Kripke frames that incorporate the order relation of intuitionistic Kripke frames and the accessibility relation of modal Kripke frames, it combines the well-known problem of looping in decision procedures based on sequent calculi for both intuitionistic and classical modal logics, which is actually exacerbated by the interactions between modalities and intuitionistic implication. With Girlando, Kuznets, Marin and Straßburger, we show that IS4 is decidable, and we conjecture that the same method can also be applied to show the decidability of IK4 [21]. We provide a constructive decision procedure, that is, given a formula it either produces a derivation which shows that the formula is valid or a finite countermodel in which the formula is false. It is built from the labelled proof system mentioned at the end of Section 1.1 [43] which inherits the advantages of labelled systems for intuitionistic propositional logic and for classical modal logic: all inference rules are invertible and there is a direct correspondence between sequents and models. The algorithm we are using to prove decidability of IS4 is presented in Chapter 6. But in Chapter 5 we use the same search algorithm and proof arguments (with the necessary changes) to prove the decision problem for other intuitionistic modal logics where transitivity is not restricting the class of frames we are working with, such as are the logics IK, IT, IKB, ITB and IK5 .

1.4 Structure of the thesis

This thesis is outlined in the following way:

In **Chapter 2** we present the syntax and semantics of classical modal logic. We introduce Hilbert and Gentzen axiomatizations, focusing on labelled sequents systems in Gentzen's.

In **Chapter 3** we follow the same structure as in Chapter 2 but for intuitionistic modal logics. We present its syntax and semantics, and we focus mainly on labelled deduction for intuitionistic modal logics.

In **Chapter 4** we provide a labelled sequent system for intuitionistic modal logics equipped with two relation symbols: the relation for intuitionistic propositional logic and the one for modal logic. We show soundness and completeness, together with an internal cut elimination proof, encompassing a wider array of intuitionistic modal logics than any existing labelled system.

In **Chapter 5** we give a decision procedure to prove decidability of the intuitionistic modal logics $\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}$ and $\mathsf{IK5}$ and we ensure its termination. We obtain this result by peforming proof search in the labelled deductive system presented in Chapter 4.

In **Chapter 6** we demonstrate, for the first time, decidability for the intuitionistic modal logic $\mathsf{IS4}$. We use the same argument as in Chapter 5, but in many definitions and proof arguments, there would be subtle differences due to the presence of transitivity.

In **Chapter 7** we present a work in progress in the formalism of deep inference using the methodology of subatomic proof theory for a fragment of classical modal logic. In particular, we present a subatomic proof system for the linear fragment of classical modal logic and we prove its cut-elimination via splitting. As a result of this proof, we prove the admissibility of the up-rules which are the cuts.

In **Chapter 8** we conclude this thesis by providing a summary of our presentations, highlighting the main contributions, and outlining potential areas for future research.

Chapter 2

Classical modal logic

In this chapter we introduce the concepts and notations that will be used in this thesis for *classical modal logic*. For a more detailed presentation we refer to the surveys by Fitting [16], Negri [50], Wansing [65] and Marin [41], on which this chapter is partly based. We start introducing the syntax and semantics of classical modal logic in Section 2.1; then we follow with Hilbert's and Gentzen's axiomatizations in Sections 2.2 and 2.3 respectively. In Gentzen's axiomatizations, we focus on labelled sequent calculus, and in particular we present the labelled sequent system introduced by Negri [49].

2.1 Syntax and semantics

The language of classical modal logic, as the so-called modal logic K, is obtained from the one of classical propositional logic by adding the modal connectives $\Box$ and $\Diamond$. Starting with a set of atomic propositions denoted a and their duals $\overline{a}$, modal formulas are constructed from the following grammar:

$$A ::= a \mid \overline{a} \mid \top \mid \bot \mid A \wedge A \mid A \vee A \mid \Box A \mid \Diamond A$$

where a stands for a propositional variable and $\overline{a}$ is its dual, and $\top$ and $\bot$ are the usual units denoting true and false respectively. In a classical setting, we always assume that formulas are in negation normal form. If we write $\overline{A}$, then it is the result of computing the De Morgan dual of connectives and atomic propositions within A, i.e. $\overline{\overline{A}} = A$, $\overline{(A \wedge B)} = \overline{A} \vee \overline{B}$ and $\overline{\Box A} = \Diamond \overline{A}$. Implication can be defined from this set of connectives by $A \supset B ::= \overline{A} \vee B$. $\top$ and $\bot$ are the usual units denoting true and false respectively.

Modal logic formulas are interpreted on relational structures. For historical reasons, such structures are known as Kripke models [34]. The semantics in terms of these models are known as *Kripke semantics* or, by the original interpretation of the modal operators, as *possible-worlds semantics*. Intuitively, a model is a directed graph with labels at the nodes, and modal formulas allow us to express properties of these graphs. We start with a graph, without considering the labels in the nodes, which it is called *frame* and then we add a mechanism to evaluate formulas which it is called a *model*:

Definition 2.1.1. A *frame* $\mathcal{F}$ is a pair $\langle W, R \rangle$ of non-empty set of worlds W and a binary accessibility relation $R \subseteq W \times W$.

Definition 2.1.2. A *model* $\mathcal{M}$ is a triple $\langle W, R, V\rangle$ with $\langle W, R\rangle$ a frame and $V\colon W \to 2^{\mathcal{A}}$ a valuation function, that is, a function mapping each world w to the subset of propositional atoms that are true at w.

Figure 2.1 shows an example of a Kripke model. We can observe a model $\mathcal{M}$ with three elements $\{w, v, u\}$. The world w is labelled with p, the worlds u is labelled with p and q, and the world v does not have a label. Formally, we have a model $\mathcal{M}{=}\langle W, R, V\rangle$ where $W = \{w, v, u\}$, $R = \{(w, v), (w, u), (v, v), (v, u), (u, v)\}$, and $V(w){=}\{p\}, V(u) = \{p, q\}$.

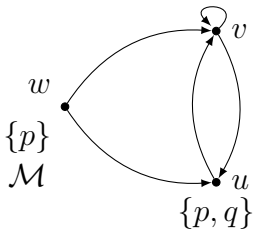


Figure 2.1: Example of a Kripke model.

The proposition expressed by a formula which involves only the usual logical connectives is determined locally in a particular world and it is independent from the state of other worlds. On the other hand, a proposition expressed by a formula which involves the modalities, depends on the state of other worlds. For instance, in a world w, the formula $\Diamond A$ expresses that the formula A is true in some world v such that v is considered *possible* from the viewpoint of w. Technically, the definition that v is *possible according to* w, it is modelled by the accessibility relation of the model. Dually, the formula $\Box A$ expresses the proposition that A is true in all the worlds v such that all of them are considered *possible* from w. In this way, the meaning of the modalities $\Box$ and $\Diamond$ is based on the primitive notion of relative truth, i.e., the truth in a world. The operators from classical modal logic describe *local* properties from the models, this means the formulas are evaluated in some specific point. In other words, the truth of a modal formula in a model $\mathcal{M}$ at a world w in a relational structure is defined as the *possible-worlds semantics* or *Kripke semantics* as follows:

$\mathcal{M}, w \Vdash a$	iff	$a \in V(w)$
$\mathcal{M}, w \Vdash \overline{a}$	iff	$a \notin V(w)$
$\mathcal{M}, w \Vdash A \wedge B$	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 $\mathcal{M}, w \Vdash B$
$\mathcal{M}, w \Vdash \Box A$	iff	for all v such that wRv, we have $\mathcal{M}, v \Vdash A$
$\mathcal{M}, w \Vdash \Diamond A$	iff	there exists a v such that wRv and $\mathcal{M}, v \Vdash A$

Notice the correspondence between $\Box$ and universal quantification $\forall$, as well as $\Diamond$ and existential quantification $\exists$. This allows us to identify classical modal logic as a fragment of first-order classical logic. We define the notions of satisfability and validity of a formula as follows:

Definition 2.1.3. A formula A is *satisfied* in a model $\mathcal{M} = \langle W, R, V \rangle$, if for every $w \in W$ we have $\mathcal{M}, w \Vdash A$. A formula A is *valid* in a frame $\mathcal{F} = \langle W, R \rangle$, if for very valuation V, the formula A is satisfied in $\langle W, R, V \rangle$.

In this thesis we are interested in studying modal logics using proof theory. There are different types of proof systems for a logical language, each with certain particular properties. For example, Hilbert systems in general consist of a set of axioms, and a few rules that allow theorems to be inferred from axioms. On the other hand, in Gentzen systems, inference rules predominate over axioms. In the next following sections, we will present both types of systems for classical modal logic.

2.2 Axiomatizations *à la Hilbert*

The basic idea behind a Hilbert-style axiomatization is to provide a set of axioms and inference rules that can be used to derive all valid formulas of a given logic. In a Hilbert-style axiomatization for modal logic, the axioms are typically designed to capture the behavior of the modal operators, such as necessity and possibility. The inference rules, on the other hand, are used to derive new formulas from existing ones using logical principles.

The language of classical modal logic is obtained from an arbitrary axiomatization of classical propositional logic by adding the following components:

- the *necessitation rule*: if A is a theorem of K then $\Box A$ is also a theorem of K;
- the *axiom of distributivity*, commonly written $\mathsf{k} ::= \Box(A \supset B) \supset (\Box A \supset \Box B)$.

Note that what we called an "axiom" in Hilbert-style axiomatizations refers to an *axiom schema*, which is a general rule rather than a specific statement. This is done to avoid having to explicitly mention the substitution rule, which is a fundamental aspect of Hilbert-style axiomatizations. Furthermore, in Hilbert's, an axiomatic proof starts from instances of the axiom schemes of classical propositional logic or of the specific modal logic in question and proceeds by application of the rules of *modus ponens* and necessitation. In particular, modus ponens is a fundamental inference rule used to derive new statements from existing ones. In other words, if we have a conditional statement of the form "if A, then B", and we know that A is valid, we can logically conclude that B must also be valid.

A *derivation* in Hilbert systems, is formed by compiling a list of formulas that either correspond to specific instances of given axiom schemes or are deduced from preceding formulas through the application of inference rules. Moreover, by incorporating the k axiom scheme in our base system, all the logics under consideration belong to the family of *normal modal logics*.

Whereas the presentation of a logic in axiomatic terms and the recognition of what qualifies as a proof in such systems are relatively straightforward, the process of actively discovering or constructing proofs is arduous and requires meticulous effort. Typically one has to start from complex instances of the axioms even to get to obvious conclusions such as $A \supset A$. For this reason, in presentations that limit themselves to axiomatic systems, as a preliminary to the actual use of such systems, a number of rules are shown *admissible*: If there is derivation of A and there is a derivation of B, then there exists a derivation of $A \wedge B$; if there is a derivation of A, then there is a derivation of $A \vee B$, and so on.

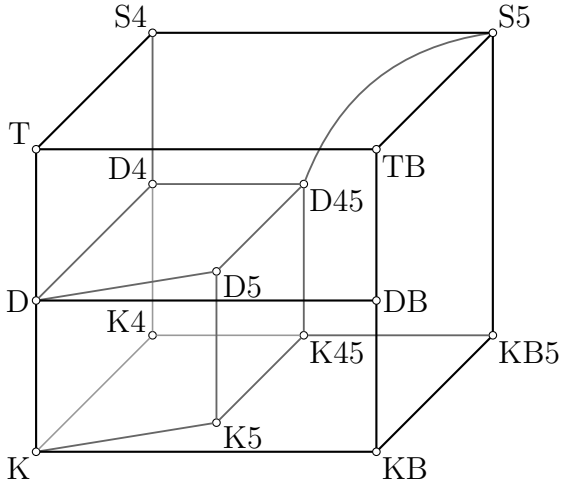


Figure 2.2: The S5-cube

On the other hand, the syntax and the semantics presented in the previous section are linked by the fact that the logic K is sound and complete with respect to the class of all frames and it is presented in the following Theorem:

Theorem 2.2.1 ([35])**.** *A formula A is a theorem of* K *if and only if A is valid in every frame.*

Furthermore, the power of this construction is that this link is not restricted to K: some classes of modal formulas correspond to specific properties of frames. An alternative way, then, to obtain modal logics stronger than K is by restricting the class of frames we want to consider, by imposing some constraints on the accessibility relation.

In this way, as modal logic developed, logicians began to explore more complex modalities and to refine their axiomatic systems to better capture the behaviors of these modalities. This led to the development of a wide range of specialized modal logics, each designed to capture specific modal phenomena in different domains. Well-known extensions of classical modal logic are obtained through the addition of one or more axioms to the system K. The table below displays the modal axioms along with the frame property they imply, as well as their corresponding first-order equivalent:

Axiom	Frame property	First-order correspondence
$\mathsf{t} : A \supset \Diamond A$	Reflexivity	$\forall x.xRx$
$\mathsf{b} : A \supset \Box\Diamond A$	Symmetry	$\forall x, y.xRy \supset yRx$
$\mathsf{d} : \Box A \supset \Diamond A$	Seriality	$\forall x \exists y.xRy$
$\mathsf{4} : \Diamond\Diamond A \supset \Diamond A$	Transitivity	$\forall x, y, z.(xRy \wedge yRz) \supset xRz$
$\mathsf{5} : \Diamond A \supset \Box\Diamond A$	Euclideaness	$\forall x, y, z.(xRy \wedge xRz) \supset yRz$
$\mathsf{2} : \Diamond\Box A \supset \Box\Diamond A$	Diamond-property	$\forall x, y, z.(xRy \wedge xRz) \supset \exists u.(yRu \wedge zRu)$

We will often consider the most common five axioms: t, b, d, 4 and 5. Picking subsets of these axioms lets us define thirty-two modal logics. For example, the sets of axioms $\{\mathsf{b}, \mathsf{4}\}$ and $\{\mathsf{t}, \mathsf{5}\}$ both yield the modal logic known as S5. On the other hand, the modal logic K4 is obtained by adding to K the axiom 4 and the logic S4 by adding the axioms t and 4.

The result of picking subsets of these common five axioms is fifteen distinct modal logics that extend each other from the modal logic K to $\mathsf{S5}$ and can be represented on the $\mathsf{S5}$-*cube* presented in Figure 2.2.

Each of the modal logics obtained from the table above has the *finite model property*: if A is not a theorem then there exists a finite model $\langle W, R, V\rangle$ such that $\mathcal{M} \not\Vdash A$. This property enables the non-theorems of the logic to be recursively enumerated. Hence, if the theorems can also be recursively enumerated (which is always the case when we possess an effective axiomatization), the ability to establish their provability becomes decidable. As a result, all the modal logics derived from the aforementioned table are decidable. For proofs of the finite model property for these logics see Chellas [11].

The axioms in question appear as specific occurrences of the axiom proposed by Lemmon and Scott in [62], well-known as *Scott-Lemmon axioms*, defined as a 4-uple of natural number $\langle h, i, j, k\rangle$ as follows:

$$\mathsf{g_{klmn}} ::= \Diamond^k \Box^l a \supset \Box^m \Diamond^n a$$

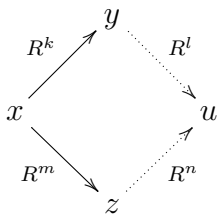
where $\Diamond^k$ denotes k occurrences of $\Diamond$ and $\Box^l$ denotes l occurrences of $\Box$. In the classical setting, the axioms $\mathsf{g_{klmn}}$ and $\mathsf{g_{mnkl}}$ are dual to each other.

The Scott-Lemmon axiom scheme has particular instances that align with the axioms previously displayed in the table in the following way:

$\mathsf{g_{0001}}$	corresponds to	$\mathsf{t} ::= a \supset \Diamond a$
$\mathsf{g_{0011}}$	corresponds to	$\mathsf{b} ::= a \supset \Box\Diamond a$
$\mathsf{g_{0101}}$	corresponds to	$\mathsf{d} ::= \Box a \supset \Diamond a$
$\mathsf{g_{1002}}$	corresponds to	$\mathsf{4} ::= \Diamond a \supset \Diamond\Diamond a$
$\mathsf{g_{1011}}$	corresponds to	$\mathsf{5} ::= \Diamond a \supset \Box\Diamond a$
$\mathsf{g_{1111}}$	corresponds to	$\mathsf{2} ::= \Diamond\Box a \supset \Box\Diamond a$

From Theorem 2.2.1 one can obtain the logic that is axiomatized by the Scott-Lemmon axiom $\mathsf{g_{klmn}}$ by examining the formulas that are valid in all frames satisfying the following confluence condition:

$$\text{for all } w, u, v \in W \text{ with } wR^k u \text{ and } wR^m v \text{ there is a } z \in W \text{ s.t. } uR^l z \text{ and } vR^n z : \quad (2.1)$$



where R^i with $i = \{k, l, m, n\}$ denotes i occurrences of R and the composition of two relations R, S on a set of worlds W is defined as usual: $R \circ S = \{(w, v) \mid \exists u.(w, u) \in R \text{ and } (u, v) \in S\}$. Then, Lemmon and Scott conclude the following theorem:

Theorem 2.2.2. *[39] Let $\mathbb{G} \subseteq \mathbb{N}$ and let G be the corresponding set of Scott-Lemmon axioms. A formula is provable in $\mathsf{K} + \mathsf{G}$ if and only if it is valid in all frames satisfying the condition 2.1 for each $\langle k, l, m, n\rangle \in \mathbb{G}$.*

We will also want to consider a particular subclass of these Scott-Lemmon axioms known as *path Scott-Lemmon axioms* [24], which corresponds to the case where $l+n$ is exactly equal to 1 (see more in Chapter 4). This property leads to the following axiom:

$$\phi_{km} ::= \Diamond^k \Box a \supset \Box^m a \tag{2.2}$$

Observe that, from the axioms listed before, $\mathsf{t}, \mathsf{b}, \mathsf{4}$ and $\mathsf{5}$ are path Scott-Lemmon axioms, but d and $\mathsf{2}$ are not.

2.3 Axiomatizations *à la Gentzen*

In Gentzen's approach, proofs are constructed by manipulating *sequents*. If the logic has De Morgan duality, we only need to consider formulas in negation normal form, i.e., negation is pushed to the atoms via the De Morgan laws as defined in Section 2.1. Then, in the classical setting, we define a sequent $\Gamma = A_1, \cdots, A_n$ as a multiset of formulas, with the comma denoting the multiset union. This sequent is nowadays called *one-sided sequent*.

As it was mentioned, while Hilbert systems have many axioms and few rules, sequent systems have few axioms and many rules. A *derivation*, denoted $\mathcal{D}$, is constructed according to these rules; it will have the structure of a tree, where each edge is a sequent and each internal node is a rule. A derivation is a *proof* of the sequent at the root, if each leaf is a rule with no premises. The height of a derivation $\mathcal{D}$, denoted by $ht(\mathcal{D})$, is the height of $\mathcal{D}$ when seen as a tree, i.e., the length of the longest path in the tree from its root to one of its leaves.

One of the crucial aspects of Gentzen's axiomatizations is their soundness and completeness in order to ensure that all valid modal statements can be formally proven. Creating a sequent system that is both sound and complete for a specific logic, involves establishing the appropriate set of sequent rules. This ensures that each theorem of the given logic can be proven within the sequent system (this is called *completeness*) and that any proof that is constructed within the sequent system yields a valid theorem of the logic (this is called *soundness*).

We say that a rule r is *admissible* for a system S if, whenever its premises are provable in S there is a proof of its conclusion in S. We say it is *derivable* if there exist a derivation in S from its premises to its conclusion, possibly using premises multiple times. One important rule to be shown admissible in a deductive system is the cut rule, which can be of this form:

$$\mathsf{cut}\ \frac{\Gamma, A \quad \overline{A}, \Gamma}{\Gamma}$$

for some formula A and its De Morgan dual $\overline{A}$. It says that if we have both Γ, A and $\overline{A}, \Gamma$, then we can derive Γ. As it was mentioned in the introduction, the cut-elimination theorem states that any theorem that has a proof in the sequent calculus using cut also has a proof without this rule. The standard proof proceeds by a double-induction on the *cut-rank* and on the height of a derivation. The rank of an instance of cut is the depth of the formula introduced by the cut reading bottom-up. The *cut-rank* of a derivation $\mathcal{D}$ is the maximal rank of a cut in $\mathcal{D}$. Demonstrating the admissibility of this rule within a specific proof system offers a degree of confidence in its suitability for backward proof-search.

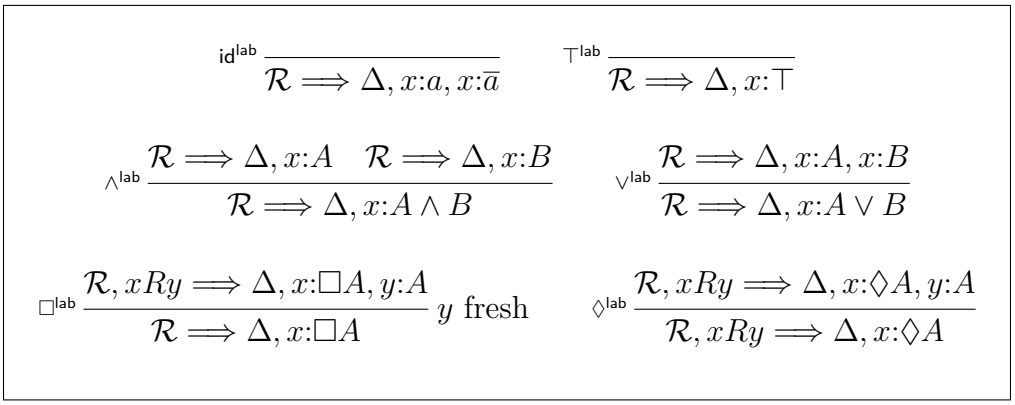


Figure 2.3: System labK

Since traditional Gentzen sequent systems for modal logic typically fail to be modular and do not satisfy most of the properties usually demanded on sequent calculus [65], alternative proof systems have been proposed in recent years for modal logic and a variety of non-classical logics. Systems that internalize the semantics of the logical constants, either implicitly through a more structured language, or explicitly through the use of labels, have been an intense object of study.

Structural proof theoretic accounts of modal logic can adopt the paradigm of *labelled deduction* in the form of labelled natural deduction and labelled sequent systems as used, for example, by Simpson [59], Vigano [63], and Negri [49]; or the one of *unlabelled deduction* in the form of sequent [4] or nested sequent systems [60] (for a survey see [41, Chapter 3]). These generalizations of the sequent framework, inspired by relational semantics, are needed to treat modalities uniformly. By extending the ordinary sequent structure with one extra element, either relational atoms between labels (referencing the accessibility relation in a Kripke model) or nested bracketing, they encode respectively graphs or trees in the sequents, giving them enough power to represent modalities. We are in particular interested in the study of modal logics using the formalism of labelled sequents.

2.3.1 Labelled sequents

Labelled deduction has been more generally proposed by Gabbay in the 80's as a unifying framework throughout proof theory in order to provide proof systems for a wide range of logics [18]. In particular, we are mostly interested in the study of the sequent presentation of these proof systems to capture classical and intuitionistic modal logics. The idea underlying what are nowadays commonly designated as labelled systems is to internalize in the calculus the explanation of modalities in terms of relational semantics.

Labelled sequents are formed by labelled formulas of the form $x{:}A$ and relational or equality atoms of the form xRy or $x = y$ respectively, where x, y range over a set of variables (called *labels*) and A is a modal formula. A (one-sided) labelled sequent is then of the form $\mathcal{R} \Longrightarrow \Delta$ where $\mathcal{R}$ denotes a set of relational or equality atoms, and Δ a multiset of labelled formulas. Following this formalism, we present in Figure 2.3.1 the labelled sequent calculus introduced by Negri [49] for classical modal logic K with the next result obtained:

Theorem 2.3.1. *[49] A formula A is a theorem of the labelled sequent system* labK *if and only if A is valid in every frame.*

As it can be observed, all the inference rules use labels in order to express explicitly the semantics of a formula in a particular state. In this way, it is reflected how the use of labels provides us with an almost direct correspondence between syntactic expressions and their semantics, making the proof theory much more intuitive. Let us see the example for $\Box$. From:

$$\mathcal{M}, x \Vdash \Box A \textit{ iff for all } y \textit{ such that } xRy \textit{ and } \mathcal{M}, y \Vdash A$$

one obtains

If y:A can be derived for an arbitrary y accessible from x, then x:$\Box A$ can be derived

and

$$\textit{If } x{:}\Box A \textit{ and } y \textit{ is accessible from } x, \textit{ then } y{:}A$$

that, since we are working on a classical setting, it is formalized by the rule

$$\Box^{\mathsf{lab}} \frac{\mathcal{R}, xRy \Longrightarrow \Delta, x{:}\Box A, y{:}A}{\mathcal{R} \Longrightarrow \Delta, x{:}\Box A}\ y \text{ fresh}$$

The principal formula x:$\Box A$ is repeated in the premiss of the rule in order to make the rule invertible. Since we are in the classical setting, the $\Diamond^{\mathsf{lab}}$-rule is the dual of the $\Box^{\mathsf{lab}}$-rule.

Chapter 3

Intuitionistic modal logic

In this chapter we introduce the concepts and notations that will be used in this thesis for *intuitionistic modal logic*. As it was presented in the classical case, we are going to introduce the syntax and semantics for intuitionistic modal logic in Section 3.1. We follow with Hilbert's and Gentzen's axiomatizations in Section 3.2 and 3.3 respectively. In Gentzen's approach, we focus on labelled sequents showing the system introduced by Simpson [59]. For a more in-depth presentation of these topics, we point to Simpson's PhD thesis [59] which continues to be the most comprehensive survey encompassing all aspects of intuitionistic modal logic.

3.1 Syntax and semantics

The language of intuitionisitic modal logic, as the so-called intuitionistic modal logic IK, is obtained from the language of intuitionistic propositional logic with the modal operators $\Box$ and $\Diamond$. Starting with a set of atomic propositions, denoted, as in the classical case, by lower case letters a, b, c,... and modal formulas, denoted by capital letters A, B, C, ..., intuitionistic modal formulas are constructed from the following grammar:

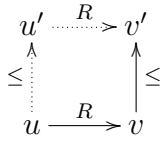
$$A ::= a \mid A \wedge A \mid A \vee A \mid \bot \mid A \supset A \mid \Box A \mid \Diamond A$$

The negation $\neg A$ is, in the intuitionistic setting, $A \supset \bot$. The implication is now a primitive connective.

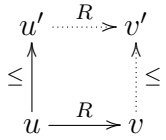
The Kripke semantics for IK was first defined by Fischer-Servi [58]. It combines the Kripke semantics for intuitionistic propositional logic and the one for classical modal logic, using two distinct relations on the set of worlds: the accessibility relation R and the preorder or future relation $\leq$. As we have done for the classical case, we define the Kripke semantics for the intuitionistic variant which now it is using a *birelational* structure as follows:

Definition 3.1.1. A *birelational frame* $\mathcal{F}$ is a triple $\langle W, R, \leq \rangle$ of a set of worlds W equipped with an accessibility relation R and a preorder relation $\leq$ (*i.e.* a reflexive and transitive relation) satisfying the following two conditions:

(F_1) For all $u, v, v' \in W$, if uRv and $v \leq v'$, there exists u' s.t. $u \leq u'$ and $u'Rv'$.



(F_2) For all $u', u, v \in W$, if $u \leq v$, there exists v' s.t. $u'Rv'$ and $v \leq v'$.



Definition 3.1.2. A *birelational model* $\mathcal{M}$ is a quadruple $\langle W, R, \leq, V\rangle$ with $\langle W, R, \leq\rangle$ a birelational frame and $V\colon W \to 2^{\mathcal{A}}$ a monotone valuation function, that is, a function mapping each world w to the subset of propositional atoms true at w, additionally subject to: if $w \leq w'$ then $V(w) \subseteq V(w')$.

Let X be an intuitionistic modal logic of the $\mathsf{S5}$-cube (see Figure 2.2)[1]. We will speak about X-birelational model when we want to be precise about the logic we are working with, and therefore, about the conditions that the model has. For example, for intuitionistic $\mathsf{S4}$ (i.e. $\mathsf{IS4}$) the conditions of transitivity and reflexivity are applied to the accessibility relation R, i.e. a $\mathsf{IS4}$-birelational model is a transitive and reflexive model.

We write $\mathcal{M}, w \Vdash a$ if $a \in V(w)$, and inductively extend the $\Vdash$ relation to all formulas, following the rules for both intuitionistic and modal Kripke models:

$$\begin{array}{rll} \mathcal{M}, w \nVdash \bot & & \\ \mathcal{M}, w \Vdash A \wedge B & \text{iff} & \mathcal{M}, w \Vdash A \text{ and } \mathcal{M}, w \Vdash B \\ \mathcal{M}, w \Vdash A \vee B & \text{iff} & \mathcal{M}, w \Vdash A \text{ or } \mathcal{M}, w \Vdash B \\ \mathcal{M}, w \Vdash A \supset B & \text{iff} & \text{for all } w' \text{ with } w \leq w', \text{ if } \mathcal{M}, w' \Vdash A \text{ then } \mathcal{M}, w' \Vdash B \\ \mathcal{M}, w \Vdash \Box A & \text{iff} & \text{for all } w' \text{ and } u \text{ with } w \leq w' \text{ and } w'Ru, \text{ we have } \mathcal{M}, u \Vdash A \\ \mathcal{M}, w \Vdash \Diamond A & \text{iff} & \text{there exists a } u \text{ such that } wRu \text{ and } \mathcal{M}, u \Vdash A. \end{array} \tag{3.1}$$

Observe that we never have that $\mathcal{M}, w \Vdash \bot$. We write $\mathcal{M}, w \nVdash A$ if it is not the case that $\mathcal{M}, w \Vdash A$, but contrarily to the classical case, we do not have $\mathcal{M}, w \Vdash \neg A$ iff $\mathcal{M}, w \nVdash A$ (since $\neg A$ is defined as $A \supset \bot$).

From the monotonicity of the valuation function V, we get a monotonicity property for the relation $\Vdash$:

Proposition 3.1.3. *(Monotonicity) For any formula A and for $w, w' \in W$, if $w \leq w'$ and $\mathcal{M}, w \Vdash A$, then $\mathcal{M}, w' \Vdash A$.*

Definition 3.1.4. A formula A is *satisfied* in a model $\mathcal{M} = \langle W, R, \leq, V\rangle$, if for all $w \in W$ we have $\mathcal{M}, w \Vdash A$. A formula A is *valid* in a frame $\mathcal{F} = \langle W, R, \leq\rangle$, if for all valuations V, the formula A is satisfied in $\langle W, R, \leq, V\rangle$.

[1]Note that the same $\mathsf{S5}$-cube is used for classical and intuitionistic approaches, this means that there are also fifteen distinct intuitionistic modal logics.

3.2 Axiomatizations *à la Hilbert*

As it was done for the classical case, the interest of having axiomatic systems that could capture the behavior of modal operators in the intuitionistic setting emerged. The axiomatization that is now generally accepted as intuitionistic modal logic IK was given by Plotkin and Stirling [52] and is equivalent to the one proposed by Fischer-Servi [58]. It is obtained from intuitionistic propositional logic IPL which is composed by the following axioms:

THEN-1:	$A \supset (B \supset A)$	THEN-2:	$(A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C)$
AND-1:	$A \wedge B \supset A$	AND-2:	$A \wedge B \supset B$
AND-3:	$A \supset (B \supset (A \wedge B))$	OR-1:	$A \supset A \vee B$
OR-2:	$B \supset A \vee B$	OR-3:	$(A \supset C) \supset ((B \supset C) \supset (A \vee B \supset C))$
FALSE:	$\bot \supset A$		

by adding the necessitation rule and five variants of the distributivity axiom k. The different consequences of the distributivity axiom are classically but not intuitionistically equivalent and they are a result of lacking De Morgan duality in the intuitionist setting:

- the *necessitation rule*: if A is a theorem then $\Box A$ is also a theorem; and
- the following five variants of k:

k_1:	$\Box(A \supset B) \supset (\Box A \supset \Box B)$	k_3:	$\Diamond(A \vee B) \supset (\Diamond A \vee \Diamond B)$	k_5:	$\Diamond\bot \supset \bot$
k_2:	$\Box(A \supset B) \supset (\Diamond A \supset \Diamond B)$	k_4:	$(\Diamond A \supset \Box B) \supset \Box(A \supset B)$		

The idea is that intuitionistic propositional logic does not allow the principle of *Excluded Middle*, so the modalities $\Box$ and $\Diamond$ no longer exhibit De Morgan duality; however, it is possible to structure the axiomatization in order to establish diverse connections between them. The most basic intuitionistic modal system one can think of would involve the exclusive consideration of the $\Box$ modality as regulated by the k axiom (or as called here k_1), which gives the system IPL + nec + k_1. However, this approach would provide no substantive insights into the behaviour of the $\Diamond$ modality. It seems that Fitch [15] was the pioneer in suggesting an approach to handle the $\Diamond$ modality in an intuitionistic system by considering the system IPL + nec + k_1 + k_2, which is now sometimes called CK for *constructive modal logic*. Wijekesera [66] also considered the axiom k_5, which states that $\Diamond$ distributes over 0-ary disjunctions, but did not assume that it would always distribute over binary disjunctions; the system he proposed was therefore IPL + nec + k_1 + k_2 + k_5. In these systems, however, the addition of the *Excluded Middle* principle does not result in classical modal logic K, this means, it is not feasible to reestablish the De Morgan duality of $\Box$ and $\Diamond$ in this scenario.

The axiomatization that is commonly recognized as *intuitionistic modal logic* denoted by IK was formulated by Plotkin and Stirling [52] and it is equivalent to the one proposed by Fischer-Servi [58] and by Ewald [13] in the case of intuitionistic tense logic. It is defined as IPL + nec + k_1 + k_2 + k_3 + k_4 + k_5.

Similarly to the classical case, the correspondence between syntax and semantics for IK can be stated as follows:

Theorem 3.2.1 ([58, 52])**.** *A formula A is a theorem of* IK *if and only if A is valid in every birelational frame.*

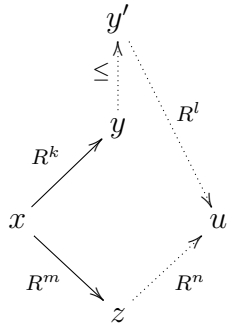


Figure 3.1: The intuitionistic klmn-incestuality condition

We now want to consider, as we have done for the classical setting, special restrictions on the birelational frames and get extensions of those basic intuitionistic modal logics, for example with the class of Scott-Lemmon axioms. Different intuitionistic modal logics are obtained through the addition of one or more variants of Scott Lemmon axioms to the system IK. However, on an intuitionistic setting, due to the lack of De Morgan duality, the axiom $\Diamond^k\Box^l a \supset \Box^m\Diamond^n a$ is no longer equivalent to $\Diamond^m\Box^n a \supset \Box^k\Diamond^l a$. So one can consider extensions with one of these two axioms, or with both in conjunction in order to capture the intuitionistic version of the corresponding classical Scott-Lemmon logic reaching to the general form of the intuitionistic Scott-Lemmon axiom:

$$\mathsf{g_{klmn}} ::= (\Diamond^k\Box^l a \supset \Box^m\Diamond^n a) \wedge (\Diamond^m\Box^n a \supset \Box^k\Diamond^l a)$$

In particular, and in the same way as in the classical case, we study the intuitionistic variants of the logics in the $\mathsf{S5}$-cube (Figure 2.2), for which we need the conjunction of the two versions of each axioms and we get the following correspondence:

$\mathsf{g_{0001}}$	corresponds to	$\mathsf{t} ::= (a \supset \Diamond a) \wedge (\Box a \supset a)$
$\mathsf{g_{0011}}$	corresponds to	$\mathsf{b} ::= (a \supset \Box\Diamond a) \wedge (\Diamond\Box a \supset a)$
$\mathsf{g_{0101}}$	corresponds to	$\mathsf{d} ::= \Box a \supset \Diamond a$
$\mathsf{g_{1002}}$	corresponds to	$\mathsf{4} ::= (\Diamond a \supset \Diamond\Diamond a) \wedge (\Box\Box a \supset \Box a)$
$\mathsf{g_{1011}}$	corresponds to	$\mathsf{5} ::= (\Diamond a \supset \Box\Diamond a) \wedge (\Diamond\Box a \supset \Box a)$
$\mathsf{g_{1111}}$	corresponds to	$\mathsf{2} ::= \Diamond\Box a \supset \Box\Diamond a$

We can then obtain stronger logics from IK extending with one or more axioms. For example, we can obtain the intuitionistic modal logic $\mathsf{IS4}$ by adding the axioms t and $\mathsf{4}$ to the logic IK, or we can get the logic $\mathsf{IK4}$ by adding only the axiom $\mathsf{4}$.

These axioms are known to obey a strong correspondence with the class of frames satisfying the condition illustrated on Figure 3.1, which we call by analogy to the classical case, *intuitionistic klmn-incestuality condition*. Plotkin and Stirling give proper correspondence results with respect to the intuitionistic setting, but they must distinguish the two dual parts of each axiom and appeal to both relations of the frames, R and $\leq$, leading to the following completeness theorem:

Theorem 3.2.2 ([52]). *An intuitionistic modal frame $\langle W, R, \leq\rangle$ validates $\Diamond^k\Box^l A \supset \Box^m\Diamond^n A$ if and only if the frame satisfies:*

if $xR^k y$ and $xR^m z$ then there exists y' such that $y \leq y'$ and there exists u such that $y'R^l u$ and $zR^n u$.

Note that the relation R^h for $h \geq 1$ is defined in the usual way: $R^1 = R$ and $R^{h+1} = R \circ R^h = \{(v, w) | \exists u \in W.\ vRu \wedge uR^h w\}$.

Fischer-Servi [58] and Simpson [59] consider the restricted family of *intuitionistic Scott-Lemmon path axioms*, where $l+n$ is exactly equal to 1. Unlike Plotkin and Stirling, they did not have to distinguish between the two relations of the frames reaching a completeness result:

Theorem 3.2.3 ([58, 59])**.** *Let* G *be a set of axiom schemes of the form:*

$$(\Diamond^k \Box a \supset \Box^m a) \wedge (\Diamond^m a \supset \Box^k \Diamond a)$$

for any natural number k, m *appearing in* G*. A formula* A *is a theorem of* IK+G *if and only if* A *is satisfied in every birelational model* $\langle W, R, \leq, V \rangle$ *that satisfies:*

$$\text{for all } w, u, v \in W \text{ if } wR^k u \text{ and } wR^m v \text{ then } uRv \tag{3.2}$$

We call the condition 3.2 satisfied in Theorem 3.2.3 *intuitionistic* SL*-path condition.* We will continue discussing about these extensions of the intuitionistic modal logic IK in Chapter 4.

3.3 Labelled sequents

In this thesis we will work mainly with what is nowadays called *two-sided sequent systems.* We can obtain these type of systems for classical modal logic but they are a more natural fit for handling intuitionistic modal logic. Two-sided systems offer the necessary flexibility to manage the intricate interactions between modal operators and the intuitionistic negation. In particular, they can explicitly represent both the antecedent (premises) and succedent (conclusions), allowing to accurately capture the intuitionistic nature of modal logic where certain classical principles might not hold as the Excluded Middle principle.

A two-sided sequent $\mathfrak{G}$ is of the form $\Gamma \Longrightarrow \Delta$ where Γ and Δ are finite (possibly empty) sets of formulas. It can also be written as $A_1, \cdots, A_n \Longrightarrow B_1, \cdots, B_n$ where the comma can still be seen as a structure connective between the formulas $A_1, \cdots, A_n$ and $B_1, \cdots, B_n$. The sequent arrow $\Longrightarrow$ denotes a consequence relation between finite sequences of formulas separated by the comma. The sets Γ and Δ are called the antecedent and the succedent of $\mathfrak{G}$, respectively.

Echoing to the definition of birelational structures, the straightforward extension of labelled deduction to the intuitionistic setting would be to use two sorts of relational atoms, one for the modal relation R and another one for the intuitionistic relation $\leq$. However, another approach was taken by Simpson [59] who followed the lines of Gentzen in a labelled context. He developed a labelled natural deduction framework for modal logics and then converted it into sequent systems with the consequent restriction to one formula on the right-hand side of each sequent. This worked as well in the labelled setting as in the ordinary sequent case. Simpson's proposed labelled sequent system for intuitionistic modal logic is representing explicitly only the accessibility relation R in the syntax and not the future relation.

In Simpson's approach, intuitionistic labelled sequents are written as $\mathcal{R}, \Gamma \Longrightarrow z{:}C$ for some multiset of labelled formulas Γ, some formula C, some label z and a set of relational

atoms $\mathcal{R}$. The rules of his labelled sequent system called labIK (presented in Figure 3.2) are very similar to the classical labelled system labK of Negri (Figure 2.3) with the standard restrictions for intuitionistic version of the right-rules $\vee^{\mathsf{lab}}_{\mathsf{R1}}$ and $\vee^{\mathsf{lab}}_{\mathsf{R2}}$ and the left-rule $\supset^{\mathsf{lab}}_{\mathsf{L}}$. Simpson obtained soundness and completeness results for the system labIK which are in Theorem 3.3.1.

$$\mathsf{id}^{\mathsf{lab}}\ \frac{}{\mathcal{R}, \Gamma, x{:}A \Longrightarrow x{:}a} \qquad \bot^{\mathsf{lab}}_{\mathsf{L}}\ \frac{}{\mathcal{R}, \Gamma, x{:}\bot \Longrightarrow z{:}A} \qquad \top^{\mathsf{lab}}_{\mathsf{R}}\ \frac{}{\mathcal{R}, \Gamma \Longrightarrow x{:}\top}$$

$$\wedge^{\mathsf{lab}}_{\mathsf{L}}\ \frac{\mathcal{R}, \Gamma, x{:}A \wedge B, x{:}A, x{:}B \Longrightarrow z{:}C}{\mathcal{R}, \Gamma, x{:}A \wedge B \Longrightarrow z{:}C} \qquad \wedge^{\mathsf{lab}}_{\mathsf{R}}\ \frac{\mathcal{R}, \Gamma \Longrightarrow x{:}A \quad \mathcal{R}, \Gamma \Longrightarrow x{:}B}{\mathcal{R}, \Gamma \Longrightarrow x{:}A \wedge B}$$

$$\vee^{\mathsf{lab}}_{\mathsf{L}}\ \frac{\mathcal{R}, \Gamma, x{:}A \Longrightarrow z{:}C \quad \mathcal{R}, \Gamma, x{:}B \Longrightarrow z{:}C}{\mathcal{R}, \Gamma, x{:}A \vee B \Longrightarrow z{:}C}$$

$$\vee^{\mathsf{lab}}_{\mathsf{R1}}\ \frac{\mathcal{R}, \Gamma \Longrightarrow x{:}A}{\mathcal{R}, \Gamma \Longrightarrow x{:}A \vee B} \qquad \vee^{\mathsf{lab}}_{\mathsf{R2}}\ \frac{\mathcal{R}, \Gamma \Longrightarrow x{:}B}{\mathcal{R}, \Gamma \Longrightarrow x{:}A \vee B}$$

$$\supset^{\mathsf{lab}}_{\mathsf{L}}\ \frac{\mathcal{R}, \Gamma, x{:}A \supset B \Longrightarrow x{:}A \quad \mathcal{R}, \Gamma, x{:}A \supset B, x{:}B \Longrightarrow z{:}C}{\mathcal{R}, \Gamma, x{:}A \supset B \Longrightarrow z{:}C} \qquad \supset^{\mathsf{lab}}_{\mathsf{R}}\ \frac{\mathcal{R}, \Gamma, x{:}A \Longrightarrow x{:}B}{\mathcal{R}, \Gamma \Longrightarrow x{:}A \supset B}$$

$$\Box^{\mathsf{lab}}_{\mathsf{L}}\ \frac{\mathcal{R}, xRy, \Gamma, x{:}\Box A, y{:}A \Longrightarrow z{:}B}{\mathcal{R}, xRy, \Gamma, x{:}\Box A \Longrightarrow z{:}B} \qquad \Box^{\mathsf{lab}}_{\mathsf{R}}\ \frac{\mathcal{R}, xRy, \Gamma \Longrightarrow y{:}A}{\mathcal{R}, \Gamma \Longrightarrow x{:}\,\Box A}\ y \text{ fresh}$$

$$\Diamond^{\mathsf{lab}}_{\mathsf{L}}\ \frac{\mathcal{R}, xRy, \Gamma, y{:}A \Longrightarrow z{:}B}{\mathcal{R}, \Gamma, x{:}\Diamond A \Longrightarrow z{:}B}\ y \text{ fresh} \qquad \Diamond^{\mathsf{lab}}_{\mathsf{R}}\ \frac{\mathcal{R}, xRy, \Gamma \Longrightarrow y{:}A}{\mathcal{R}, xRy, \Gamma \Longrightarrow x{:}\Diamond A}$$

Figure 3.2: System labIK

Theorem 3.3.1 (Simpson [59])**.** *A formula A is provable in the calculus* labIK *if and only if A is valid in every birelational frame.*

In [59], Simpson extends his basic sequent system for IK to the geometric axiom family [48]. The goal of this family of axioms is to consider the logic as defined by a given class of frames. In particular, they generalize Scott-Lemmon frame properties, which were investigated by Simpson for intuitionistic modal logics. For example, you can extend Simpson's sequent system with the following rule:

$$\boxtimes_4\ \frac{\mathcal{R}, wRv, vRu, wRu, \Gamma \Longrightarrow \Delta}{\mathcal{R}, wRv, vRu, \Gamma \Longrightarrow \Delta}\ u' \text{ fresh}$$

and obtain a sound and complete system wrt. IK plus the axiom 4: $(\Diamond\Diamond A \supset \Diamond A) \wedge (\Box A \supset \Box\Box A)$, that is, wrt. to all frames in which the accessibility relation R is transitive.

In [52], Plotkin and Stirling give a more general correspondence result than Theorem 3.2.1, that is, for intuitionistic modal logic extended with a family of axioms wrt. some classes of birelational frames. For example, the frames that validate the axiom $\mathsf{4}_\Diamond$: $\Diamond\Diamond A \supset \Diamond\Diamond A$ are exactly the ones satisfying the condition:

($\oplus_{\mathsf{4}}$) if wRv and vRu, there exists a $u' \in W$ s.t. $u \leq u'$ and wRu'.

In the next chapter, we show that incorporating the preorder symbol into the syntax of our sequents allows us to also obtain a sound and complete proof system for the intuitionistic modal logic extended with axiom $\mathsf{4}_\Diamond$, by designing the following rule:

$$\oplus_4 \frac{\mathcal{R}, wRv, vRu, u{\leq}u', wRu', \Gamma \Longrightarrow \Delta}{\mathcal{R}, wRv, vRu, \Gamma \Longrightarrow \Delta}\ u' \text{ fresh}$$

Therefore, we decompose further the formalism of labelled sequents and extend the reach of labelled deduction to the logics studied in [52]. Let us continue with the details of this new proposal to labelled sequents in the next chapter.

Chapter 4

Fully labelled proof system

In this chapter, we develop a *fully labelled proof system* for intuitionistic modal logics such that there is not only one, but two relation symbols appearing in sequents: one for the accessibility relation associated with the Kripke semantics for normal modal logics and one for the preorder relation associated with the Kripke semantics for intuitionistic logic. This puts our system in close correspondence with the standard birelational Kripke semantics for intuitionistic modal logics introduced in Chapter 3, Section 3.1.

We show in Section 4.1 our motivation and interest to study this extension of labelled deduction to the intuitionistic setting. Then we present the system in Section 4.2 and completeness and soundness results in Sections 4.3 and 4.4 respectively. We also show an internal cut elimination proof for our system in Section 4.5. As a result of explicitly incorporating both relations into the system, it can encompass a wider range of intuitionistic modal logics than other existing labelled systems [59]. We offer a detailed exploration of these extensions in Section 4.6.

4.1 Motivation

Echoing the definition of birelational structures, we consider an extension of labelled deduction to the intuitionistic setting. Since their introduction in the 1980s by Gabbay [18], *labelled proof calculi* have been widely used by proof theorists to give sound, complete and cut-free deductive systems to a broad range of logics. Unlike hypersequents [3], nested sequents [30, 8, 53], 2-sequents [46], or linear nested sequents [38], labelled calculi have the advantage of being more uniform and being able to accommode a larger class of logics. This is one of the reasons why we are interested in studying this formalism in the intuitionistic setting to be able to study extensions of IK like intuitionistic modal logic $\mathsf{IS4}$ and $\mathsf{IK4}$. We will come back to this in Chapter 6.

As it was mentioned in Chapter 2, standard labelled sequent calculi attach to every formla A a label x, witten as $x{:}A$, and additionally use *relational atoms* of the form xRy where R is a binary relation symbol. These calculi work best for logics with standard Kripke semantics, as in this case the relation R is used to encode the accessibility relation in the Kripke models, and the frame conditions corresponding to the desired logic can be directly encoded as inference rules. Prominent examples are classical modal logics and intuitionistic propositional logic, where, for example, the frame condition of transitivity

$(\forall xyz.\, xRy \wedge yRz \supset xRz)$, can be straightforwardly translated into the inference rule

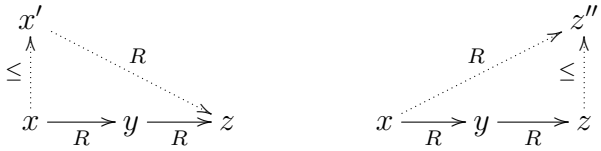
$$\mathit{Rtr}\,\frac{\mathcal{R}, xRy, yRz, xRz, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xRy, yRz, \Gamma \Longrightarrow \Delta} \tag{4.1}$$

where $\mathcal{R}$ stands for a set of relational atoms, and Γ and Δ for multi-sets of labelled formulas.

However, in this chapter we are concerned with *intuitionistic modal logics*, whose Kripke semantics is based on *birelational* frames, i.e., they have two binary relations instead of one: one relation R that corresponds to the accessibility relation in Kripke frames for modal logics, and a relation $\leq$ that corresponds to the preorder relation in Kripke frames for intuitionistic logic. Consequently, standard labelled systems for these logics have certain shortcomings:

1. The transitivity rule in (4.1) can be axiomatized by the conjunction of the two versions of the 4-axiom as it was presented in Chapter 3. However, in intuitionistic modal logic they are not equivalent, and even though some logics (like IK4 and IS4) contain both axioms, they can also be added independently to the logic IK. The proof theory of these new logics has not been studied before; no existing labelled (or label-free) proof system can handle them, even though the corresponding frame conditions

   $$\forall xyz.\, xRy \wedge yRz \supset (\exists x'.\, x \leq x' \wedge x'Rz) \quad \text{and} \quad \forall xyz.\, xRy \wedge yRz \supset (\exists z'.\, z \leq z' \wedge xRz')\,, \tag{4.2}$$

   respectively, have already been studied in [52].

2. The correspondence between the syntax and the semantics is not as clean as one would expect. As only the R-relation (and not the $\leq$-relation) of the frame is visible in an ordinary labelled sequent, we only have that a sequent Γ is provable if and only if is satisfied in all *graph-consistent*[1] models, as already observed by Simpson in his PhD thesis [59] and considered as an inelegant solution (see also [45]).

In order to address these two concerns we propose here to enrich usual labelled sequents by allowing both, relational atoms of the form $x{\leq}y$ and of the form xRy. Consequently, we can easily translate the frame conditions in (4.2) into inference rules as the following two:

$$4^{\Box}\,\frac{\mathcal{R}, xRy, yRz, x'Rz, x{\leq}x', \Gamma \Longrightarrow \Delta}{\mathcal{R}, xRy, yRz, \Gamma \Longrightarrow \Delta}\; x' \text{ fresh} \qquad \text{and} \qquad 4^{\Diamond}\,\frac{\mathcal{R}, xRy, yRz, xRz', z{\leq}z', \Gamma \Longrightarrow \Delta}{\mathcal{R}, xRy, yRz, \Gamma \Longrightarrow \Delta}\; z' \text{ fresh}$$

This allows us to define cut-free deductive systems for a wide range of logics that could not be treated before. Furthermore, the relation between syntax and semantics is as one would expect: A sequent is provable in our system if and only if it is valid in all models.

Besides that, there is another pleasant observation to make about our system. Ordinary labelled sequent systems for intuitionistic modal logic are single-conclusion [59] as Simpson's

[1] This means that every layer in the model can be lifted to any future of any world in that layer. See [59] and [45] for a formal definition and discussion.

system presented in Chapter 3, Section 3.3. The same is true for the corresponding nested sequent systems [60, 44]. It is possible to express Maehara style multiple-conclusion systems in nested sequents [61], and therefore also in ordinary labelled sequents. However, also in these systems there are rules ($\supset^\circ$ and $\Box^\circ$) that force a single-conclusion premise.Maffezioli, Naibo and Negri have considered in [40] a labelled system for intuitionistic bimodal epistemic logic which is multi-conclusion. Our system uses the same principle, both labelled and multi-conclusion sequents, but we use it in a more general setting and extend it to a framework for many intuitionistic modal logics. This eliminates the undesired discrepancy as, consequently, every rule in our system is invertible, which means that we never delete information in a bottom-up proof search. As a result, we present our fully labelled proof system to capture intuitionistic modal logics which was also useful to continue our research on the decision problems for logics of our interest as $\mathsf{IS4}$ and $\mathsf{IK4}$ (see Chapter 6).

4.2 The proof system

Given a two-sided labelled sequent $\mathcal{R}, \Gamma \Longrightarrow \Delta$, where $\mathcal{R}$ is a set of relational atoms and Γ and Δ are multi-sets of labelled formulas, we obtain the inferece rules for system $\mathsf{labIK}_{\leq}$ for the intuitionistic modal logic IK as it is presented in Figure 4.1.

As it can be observed, the inference rules are in close correspondence with the standard birelational Kripke semantics for intuitionistic modal logics presented in the previous chapter in (3.1). As it was mentioned, we enrich usual labelled sequents by allowing both, relational atoms of the form xRy (capturing the accessibility relation R) and of the form $x{\leq}y$ (capturing the preorder relation $\leq$). In the Kripke semantics, the two connectives $\supset$ and $\Box$ are the ones that make use of the preorder relation $\leq$. This relation is reflexive and transitive and in order to capture that in the proof system, we need to add the inference rules $\leq\mathsf{rf}$ and $\leq\mathsf{tr}$. These can be obtained by applying the axioms-as-rules methodology as in [40] and we get the following rules which are part of our system:

$$\leq\mathsf{rf}\ \frac{\mathcal{R}, x \leq x, \Gamma \Longrightarrow \Delta}{\mathcal{R}, \Gamma \Longrightarrow \Delta} \qquad \leq\mathsf{tr}\ \frac{\mathcal{R}, x \leq y, y \leq z, x \leq z, \Gamma \Longrightarrow \Delta}{\mathcal{R}, x \leq y, y \leq z, \Gamma \Longrightarrow \Delta}$$

On the other hand, in the semantics, the two relations R and $\leq$ are strongly connected through the two conditions $\mathsf{F_1}$ and $\mathsf{F_2}$. These need to be reflected at the level of the proof system, which is done by the two rules $\mathsf{F_1}$ and $\mathsf{F_2}$:

$$\mathsf{F_1}\ \frac{\mathcal{R}, xRy, y{\leq}z, x{\leq}u, uRz, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xRy, y{\leq}z, \Gamma \Longrightarrow \Delta}\ u \text{ fresh} \qquad \mathsf{F_2}\ \frac{\mathcal{R}, xRy, x{\leq}z, y{\leq}u, zRu, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xRy, x{\leq}z, \Gamma \Longrightarrow \Delta}\ u \text{ fresh}$$

Another indication of the fact that $\mathsf{labIK}_{\leq}$ is well-designed, is that the general identity axiom is admissible. We show this in the following Proposition:

Proposition 4.2.1. *The following general identity axiom* $\mathsf{id_g}\ \dfrac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \Longrightarrow \Delta, y{:}A}$ *is admissible for* $\mathsf{labIK}_{\leq}$.

Proof. As standard, we proceed by structural induction on A. The two base cases $A = a$ and $A = \bot$ and the inductive cases are shown below:

$$\mathsf{id}\ \frac{}{\mathcal{R}, x \leq y, \Gamma, x{:}a \Longrightarrow \Delta, y{:}a} \qquad \bot^{\bullet}\ \frac{}{\mathcal{R}, \Gamma, x{:}\bot \Longrightarrow \Delta}$$

$$\wedge^{\bullet}\ \frac{\mathcal{R}, \Gamma, x{:}A, x{:}B \Longrightarrow \Delta}{\mathcal{R}, \Gamma, x{:}A \wedge B \Longrightarrow \Delta} \qquad \wedge^{\circ}\ \frac{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \quad \mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}B}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \wedge B}$$

$$\vee^{\bullet}\ \frac{\mathcal{R}, \Gamma, x{:}A \Longrightarrow \Delta \quad \mathcal{R}, \Gamma, x{:}B \Longrightarrow \Delta}{\mathcal{R}, \Gamma, x{:}A \vee B \Longrightarrow \Delta} \qquad \vee^{\circ}\ \frac{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A, x{:}B}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \vee B}$$

$$\supset^{\circ}\ \frac{\mathcal{R}, x \leq y, \Gamma, y{:}A \Longrightarrow \Delta, y{:}B}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \supset B}\ y \text{ fresh}$$

$$\supset^{\bullet}\ \frac{\mathcal{R}, x \leq y, x{:}A \supset B, \Gamma \Longrightarrow \Delta, y{:}A \quad \mathcal{R}, x \leq y, \Gamma, y{:}B \Longrightarrow \Delta}{\mathcal{R}, x \leq y, \Gamma, x{:}A \supset B \Longrightarrow \Delta}$$

$$\Box^{\bullet}\ \frac{\mathcal{R}, x \leq y, yRz, \Gamma, x{:}\Box A, z{:}A \Longrightarrow \Delta}{\mathcal{R}, x \leq y, yRz, \Gamma, x{:}\Box A \Longrightarrow \Delta} \qquad \Box^{\circ}\ \frac{\mathcal{R}, x \leq y, yRz, \Gamma \Longrightarrow \Delta, z{:}A}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}\Box A}\ y, z \text{ fresh}$$

$$\Diamond^{\bullet}\ \frac{\mathcal{R}, xRy, \Gamma, y{:}A \Longrightarrow \Delta}{\mathcal{R}, \Gamma, x{:}\Diamond A \Longrightarrow \Delta}\ y \text{ fresh} \qquad \Diamond^{\circ}\ \frac{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A, y{:}A}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A}$$

$$\leq\!\mathsf{rf}\ \frac{\mathcal{R}, x \leq x, \Gamma \Longrightarrow \Delta}{\mathcal{R}, \Gamma \Longrightarrow \Delta} \qquad \leq\!\mathsf{tr}\ \frac{\mathcal{R}, x \leq y, y \leq z, x \leq z, \Gamma \Longrightarrow \Delta}{\mathcal{R}, x \leq y, y \leq z, \Gamma \Longrightarrow \Delta}$$

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

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

Figure 4.1: System $\mathsf{labIK}_{\leq}$

- Base cases:

$$\mathsf{id_g}\ \frac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}a \Longrightarrow \Delta, y{:}a} \rightsquigarrow \mathsf{id}\ \frac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}a \Longrightarrow \Delta, y{:}a}$$

$$\mathsf{id_g}\ \frac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}\bot \Longrightarrow \Delta, y{:}\bot} \rightsquigarrow \bot^{\bullet}\ \frac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}\bot \Longrightarrow \Delta, y{:}\bot}$$

- $A \wedge B$:

$$\mathsf{id_g}\ \frac{}{\mathcal{R}, x{\leq}y; \Gamma, x{:}A \wedge B \Longrightarrow \Delta, y{:}A \wedge B} \rightsquigarrow$$

$$\dfrac{\dfrac{\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A, x{:}B \Longrightarrow \Delta, y{:}A}}{\wedge^\circ}\quad \dfrac{\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A, x{:}B \Longrightarrow \Delta, y{:}B}}{\wedge^\circ}}{\dfrac{\mathcal{R}, x{\leq}y, \Gamma, x{:}A, x{:}B \Longrightarrow \Delta, y{:}A \wedge B}{\wedge^\bullet}}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \wedge B \Longrightarrow \Delta, y{:}A \wedge B}$$

- $A \vee B$:

$$\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y; \Gamma, x{:}A \vee B \Longrightarrow \Delta, y{:}A \vee B} \quad \rightsquigarrow$$

$$\dfrac{\dfrac{\dfrac{\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \Longrightarrow \Delta, y{:}A}}{\vee^\circ}}{\vee^\bullet}\dfrac{\dfrac{\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}B \Longrightarrow \Delta, y{:}B}}{\vee^\circ}}{\vee^\bullet}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \vee B \Longrightarrow \Delta, y{:}A \vee B}$$

- $A \supset B$:

$$\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \supset B \Longrightarrow \Delta, y{:}A \supset B} \quad \rightsquigarrow$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y, y{\leq}z, x{\leq}z, z{\leq}z, \Gamma, x{:}A \supset B, z{:}A \Longrightarrow \Delta, z{:}B, z{:}A}}{\leq\mathsf{rf}}}{\supset^\bullet}\dfrac{\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y, y{\leq}z, x{\leq}z, z{\leq}z, \Gamma, z{:}B, z{:}A \Longrightarrow \Delta, z{:}B}}{\leq\mathsf{rf}}}{\leq\mathsf{tr}}}{\supset^\circ}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \supset B \Longrightarrow \Delta, y{:}A \supset B}\ z \text{ fresh}$$

- $\Box A$:

$$\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}\Box A \Longrightarrow \Delta, y{:}\Box A} \quad \rightsquigarrow$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y, y{\leq}z, x{\leq}z, zRw, w{\leq}w, \Gamma, z{:}\Box A, w{:}A \Longrightarrow \Delta, w{:}A}}{\leq\mathsf{rf}}}{\Box^\bullet}}{\leq\mathsf{tr}}}{\Box^\circ}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}\Box A \Longrightarrow \Delta, y{:}\Box A}\ z, w \text{ fresh}$$

- $\Diamond A$:

$$\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y; \Gamma, x{:}\Diamond A \Longrightarrow \Delta, y{:}\Diamond A} \quad \rightsquigarrow$$

$$\dfrac{\dfrac{\dfrac{\mathsf{id_g}\,\dfrac{}{\mathcal{R}, x{\leq}y, xRz, z{\leq}u, yRu, \Gamma, z{:}A \Longrightarrow \Delta, y{:}\Diamond A, u{:}A}}{\Diamond^\circ}}{\mathsf{F}_2}}{\Diamond^\bullet}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}\Diamond A \Longrightarrow \Delta, y{:}\Diamond A}\ u \text{ fresh},\ z \text{ fresh}$$

□

We can see in this proof the need of the rules $\leq$rf and $\leq$tr in the cases $A \supset B$ and $\Box A$. Furthermore, the need of the rule F_2 is shown in the case of $\Diamond A$. In the next section we will show the need of the rule F_1 and we will see that the rules $\leq$rf and $\leq$tr are also needed in other situations.

In the following sections, we will show that the system $\mathsf{labIK}_{\leq}$ is sound and complete. For the completeness proof we proceed via cut elimination. The cut rule has the following shape:

$$\mathsf{cut}\ \frac{\mathcal{R}, \Gamma \Longrightarrow \Delta, z{:}C \qquad \mathcal{R}, \Gamma, z{:}C \Longrightarrow \Delta}{\mathcal{R}, \Gamma \Longrightarrow \Delta} \tag{4.3}$$

Then we can summarize soundness, completeness, and cut admissibility of $\mathsf{labIK}_{\leq}$ in the following Theorem:

Theorem 4.2.2. *For any formula A, the following are equivalent.*

1. *A is a theorem of* IK.
2. *A is provable in* $\mathsf{labIK}_{\leq} + \mathsf{cut}$.
3. *A is provable in* $\mathsf{labIK}_{\leq}$.
4. *A is valid in every bi-relational frame.*

The proof of this theorem is the topic of the following sections. The equivalence of 1 and 4 has already been stated in Theorem 3.2.1 [58, 52]. The implication $1 \Longrightarrow 2$ (i.e. completeness) is shown in Section 4.3, the implication $2 \Longrightarrow 3$ (i.e. cut elimination) is shown in Section 4.5, and finally, the implication $3 \Longrightarrow 4$ (i.e. soundness) is shown in Section 4.4.

Once we have shown cut elimination (the implication $2 \Longrightarrow 3$ of Theorem 4.2.2), we can show that the following rules for *monotonicity* are admissible in our system:

$$\mathsf{mon}^{\bullet}\ \frac{\mathcal{R}, x \leq y, \Gamma, x{:}A, y{:}A \Longrightarrow \Delta}{\mathcal{R}, x \leq y\Gamma, x{:}A \Longrightarrow \Delta} \qquad\qquad \mathsf{mon}^{\circ}\ \frac{\mathcal{R}, x \leq y, \Gamma \Longrightarrow \Delta, x{:}A, y{:}A}{\mathcal{R}, x \leq y, \Gamma \Longrightarrow \Delta, y{:}A} \tag{4.4}$$

Since these rules are a form of contraction, it would cause the same problems as contraction in a cut elimination proof. Hence, it is preferable to have a system in which these rules are admissible. This is the reason why we have monotonicity incorporated in the rules id, $\supset^{\bullet}$ and $\Box^{\bullet}$ in Figure 4.1. Then, we can show admissibility of the rules $\mathsf{mon}^{\bullet}$ and mon° in the following Proposition:

Proposition 4.2.3. *The rules* $\mathsf{mon}^{\bullet}$ *and* mon° *are admissible for* $\mathsf{labIK}_{\leq}$.

Proof. The rule $\mathsf{mon}^{\bullet}$ can be derived using the general identity id_{g} and the cut rule:

$$\mathsf{cut}\ \frac{\mathsf{id}_{\mathsf{g}}\ \dfrac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \Longrightarrow \Delta, y{:}A} \qquad \mathcal{R}, x{\leq}y, \Gamma, x{:}A, y{:}A \Longrightarrow \Delta}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \Longrightarrow \Delta}$$

and both these rules are admissible by Proposition 4.2.1 and Theorem 4.2.2. The case for mon° is similar:

$$\mathsf{cut}\,\frac{\mathcal{R}, x{\leq}y, \Gamma \Longrightarrow \Delta, x{:}A, y{:}A \qquad \mathsf{id_g}\,\overline{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \Longrightarrow \Delta, y{:}A}}{\mathcal{R}, x{\leq}y, \Gamma \Longrightarrow \Delta, y{:}A}$$

□

Remark 4.2.4. As mentioned above, the monotonicity rules $\mathsf{mon}^\bullet$ and mon° are a form of contraction. So, it is not a surprise that the contraction rules

$$\mathsf{cont}^\bullet\,\frac{\mathcal{R}, \Gamma, x{:}A, x{:}A \Longrightarrow \Delta}{\mathcal{R}, \Gamma, x{:}A \Longrightarrow \Delta} \qquad \mathsf{cont}^\circ\,\frac{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A, x{:}A}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A}$$

are admissible in our system, as they are derivable as follows:

$$\mathsf{mon}^\bullet\,\frac{\mathcal{R}, x{\leq}x, \Gamma, x{:}A, x{:}A \Longrightarrow \Delta}{\leq\mathsf{rf}\,\dfrac{\mathcal{R}, x{\leq}x, \Gamma, x{:}A \Longrightarrow \Delta}{\mathcal{R}, \Gamma, x{:}A \Longrightarrow \Delta}} \qquad \mathsf{mon}^\circ\,\frac{\mathcal{R}, x{\leq}x, \Gamma \Longrightarrow \Delta, x{:}A, x{:}A}{\leq\mathsf{rf}\,\dfrac{\mathcal{R}, x{\leq}x, \Gamma \Longrightarrow \Delta, x{:}A}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A}}$$

4.3 Completeness

In this section we show our system at work, as most of the section consists of derivations of axioms of the intuitionistic modal logic IK in the fully labelled system $\mathsf{labIK}_{\leq}$. More precisely, we prove completeness of $\mathsf{labIK}_{\leq} + \mathsf{cut}$. Intuitively, this property indicates that our system has everything that is necessary to show the truths of the characterizing logic. We have seen already in the proof of Proposition 4.2.1 the need of the rules $\mathsf{F_2}$, $\leq\mathsf{rf}$ and $\leq\mathsf{tr}$. In the following proof of completeness of $\mathsf{labIK}_{\leq} + \mathsf{cut}$, we also see the need of the rule $\mathsf{F_1}$, as well as the need again of the rules $\leq\mathsf{rf}$ and $\leq\mathsf{tr}$. This is the proof for the implication $1 \Longrightarrow 2$ of Theorem 4.2.2, which is stated again below:

Theorem 4.3.1. *For any formula A, if A is a theorem of the intuitionistic modal logic IK then A is provable in $\mathsf{labIK}_{\leq} + \mathsf{cut}$.*

Proof. We prove completeness of our system $\mathsf{labIK}_{\leq}$ with respect to the Hilbert system presented in Chapter 3, Section 3.2. In order to get this proof, we need to prove the following:

- ⋆ prove that the five variants of the axiom of distributivity $\mathsf{k_1}, ..., \mathsf{k_5}$ are proved in system $\mathsf{labIK}_{\leq}$.
- ⋆ prove that all the axioms of intuitionistic propositional logic presented in Section 3.2 are proved in system $\mathsf{labIK}_{\leq}$.
- ⋆ simulate modus ponens and necessitation rules.

We begin by showing how the axioms $\mathsf{k_1}$–$\mathsf{k_5}$ are proved in system $\mathsf{labIK}_{\leq}$.

- k_1:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{R}, y{:}\Box(A \supset B), z{:}\Box A, u{:}A, u{:}A \supset B \Longrightarrow u{:}B, u{:}A}\,\mathsf{id_g}}{\mathcal{R}, y{:}\Box(A \supset B), z{:}\Box A, u{:}A, u{:}A \supset B, u{:}B \Longrightarrow u{:}B}\,\mathsf{id_g}}{x{\leq}y, y{\leq}z, z{\leq}w, y{\leq}w, u{\leq}u, wRu, y{:}\Box(A \supset B), z{:}\Box A, u{:}A, u{:}A \supset B \Longrightarrow u{:}B}\supset^{\bullet}}{x{\leq}y, y{\leq}z, z{\leq}w, y{\leq}w, wRu, y{:}\Box(A \supset B), z{:}\Box A, u{:}A, u{:}A \supset B \Longrightarrow u{:}B}\leq_{\mathsf{rf}}}{x{\leq}y, y{\leq}z, z{\leq}w, y{\leq}w, wRu, y{:}\Box(A \supset B), z{:}\Box A, u{:}A \Longrightarrow u{:}B}\Box^{\bullet}}{x{\leq}y, y{\leq}z, z{\leq}w, y{\leq}w, wRu, y{:}\Box(A \supset B), z{:}\Box A, u{:}A \Longrightarrow u{:}B}\leq_{\mathsf{tr}}}{x{\leq}y, y{\leq}z, z{\leq}w, wRu, y{:}\Box(A \supset B), z{:}\Box A \Longrightarrow u{:}B}\Box^{\bullet}}{x{\leq}y, y{\leq}z, y{:}\Box(A \supset B), z{:}\Box A \Longrightarrow z{:}\Box B}\Box^{\circ}\ w, u \text{ fresh}}{x{\leq}y, y{:}\Box(A \supset B) \Longrightarrow y{:}\Box A \supset \Box B}\supset^{\circ}\ z \text{ fresh}}{\Longrightarrow x{:}\Box(A \supset B) \supset (\Box A \supset \Box B)}\supset^{\circ}\ y \text{ fresh}$$

  where $\mathcal{R}$ is equal to: $x{\leq}y, y{\leq}z, z{\leq}w, y{\leq}w, u{\leq}u, wRu$.

- k_2:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{R}, y{:}\Box(A \supset B), u{:}A, u{:}A \supset B \Longrightarrow z{:}\Diamond B, u{:}B, u{:}A}\,\mathsf{id_g}}{\mathcal{R}, y{:}\Box(A \supset B), u{:}A, u{:}A \supset B, u{:}B \Longrightarrow z{:}\Diamond B, u{:}B}\,\mathsf{id_g}}{x{\leq}y, y{\leq}z, zRu, u{\leq}u, y{:}\Box(A \supset B), u{:}A, u{:}A \supset B \Longrightarrow z{:}\Diamond B, u{:}B}\supset^{\bullet}}{x{\leq}y, y{\leq}z, zRu, y{:}\Box(A \supset B), u{:}A, u{:}A \supset B \Longrightarrow z{:}\Diamond B, u{:}B}\leq_{\mathsf{rf}}}{x{\leq}y, y{\leq}z, zRu, y{:}\Box(A \supset B), u{:}A \Longrightarrow z{:}\Diamond B, u{:}B}\Box^{\bullet}}{x{\leq}y, y{\leq}z, zRu, y{:}\Box(A \supset B), u{:}A \Longrightarrow z{:}\Diamond B}\Diamond^{\circ}}{x{\leq}y, y{\leq}z, y{:}\Box(A \supset B), z{:}\Diamond A \Longrightarrow z{:}\Diamond B}\Diamond^{\bullet}\ u \text{ fresh}}{x{\leq}y, y{:}\Box(A \supset B) \Longrightarrow y{:}(\Diamond A \supset \Diamond B)}\supset^{\circ}\ z \text{ fresh}}{\Longrightarrow x{:}\Box(A \supset B) \supset (\Diamond A \supset \Diamond B)}\supset^{\circ}\ y \text{ fresh}$$

  where $\mathcal{R}$ is equal to $x{\leq}y, y{\leq}z, zRu, u{\leq}u$.

- k_3:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x{\leq}y, z{\leq}z, yRz, z{:}A \Longrightarrow y{:}\Diamond A, z{:}A, y{:}\Diamond B}\,\mathsf{id_g}}{x{\leq}y, yRz, z{:}A \Longrightarrow y{:}\Diamond A, z{:}A, y{:}\Diamond B}\leq_{\mathsf{rf}}}{x{\leq}y, yRz, z{:}A \Longrightarrow y{:}\Diamond A, y{:}\Diamond B}\Diamond^{\circ}}{x{\leq}y, yRz, z{:}A \Longrightarrow y{:}\Diamond A \vee \Diamond B}\vee^{\circ}}{x{\leq}y, yRz, z{:}A \vee B \Longrightarrow y{:}\Diamond A \vee \Diamond B}\vee^{\bullet}\qquad \dfrac{\dfrac{\dfrac{\dfrac{}{x{\leq}y, z{\leq}z, yRz, z{:}B \Longrightarrow y{:}\Diamond A, y{:}\Diamond B, z{:}B}\,\mathsf{id_g}}{x{\leq}y, yRz, z{:}B \Longrightarrow y{:}\Diamond A, y{:}\Diamond B, z{:}B}\leq_{\mathsf{rf}}}{x{\leq}y, yRz, z{:}B \Longrightarrow y{:}\Diamond A, y{:}\Diamond B}\Diamond^{\circ}}{x{\leq}y, yRz, z{:}B \Longrightarrow y{:}\Diamond A \vee \Diamond B}\vee^{\circ}}{x{\leq}y, yRz, z{:}A \vee B \Longrightarrow y{:}\Diamond A \vee \Diamond B}\Diamond^{\bullet}\ z \text{ fresh}}{x{\leq}y, y{:}\Diamond(A \vee B) \Longrightarrow y{:}\Diamond A \vee \Diamond B}\supset^{\circ}\ y \text{ fresh}}{\Longrightarrow x{:}\Diamond(A \vee B) \supset (\Diamond A \vee \Diamond B)}$$

- k_4:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{R}, u{\leq}u, y{:}\Diamond A \supset \Box B, u{:}A \Longrightarrow u{:}B, t{:}\Diamond A, u{:}A}\,\mathsf{id_g}}{\mathcal{R}, y{:}\Diamond A \supset \Box B, u{:}A \Longrightarrow u{:}B, t{:}\Diamond A, u{:}A}\,{\leq}\mathsf{rf}}{\mathcal{R}, y{:}\Diamond A \supset \Box B, u{:}A \Longrightarrow u{:}B, t{:}\Diamond A}\,\Diamond^{\circ} \quad \dfrac{\dfrac{\dfrac{}{\mathcal{R}, t{\leq}t, u{\leq}u, y{:}\Diamond A \supset \Box B, u{:}A, t{:}\Box B, u{:}B \Longrightarrow u{:}B}\,\mathsf{id_g}}{\mathcal{R}, t{\leq}t, y{:}\Diamond A \supset \Box B, u{:}A, t{:}\Box B, u{:}B \Longrightarrow u{:}B}\,{\leq}\mathsf{rf}}{\mathcal{R}, t{\leq}t, y{:}\Diamond A \supset \Box B, u{:}A, t{:}\Box B \Longrightarrow u{:}B}\,\Box^{\bullet}}{\mathcal{R}, y{:}\Diamond A \supset \Box B, u{:}A, t{:}\Box B \Longrightarrow u{:}B}\,\supset^{\bullet}}{x{\leq}y, y{\leq}z, w{\leq}u, z{\leq}t, y{\leq}t, zRw, tRu, y{:}\Diamond A \supset \Box B, u{:}A \Longrightarrow u{:}B}\,{\leq}\mathsf{tr}}{x{\leq}y, y{\leq}z, w{\leq}u, z{\leq}t, zRw, tRu, y{:}\Diamond A \supset \Box B, u{:}A \Longrightarrow u{:}B}\,\mathsf{F}_1\ t \text{ fresh}}{x{\leq}y, y{\leq}z, w{\leq}u, zRw, y{:}\Diamond A \supset \Box B, u{:}A \Longrightarrow u{:}B}\,\supset^{\circ}\ u \text{ fresh}}{x{\leq}y, y{\leq}z, zRw, y{:}\Diamond A \supset \Box B \Longrightarrow w{:}A \supset B}\,\Box^{\circ}\ z, w \text{ fresh}}{x{\leq}y, y{:}\Diamond A \supset \Box B \Longrightarrow y{:}\Box(A \supset B)}\,\supset^{\circ}\ y \text{ fresh}}{\Longrightarrow x{:}(\Diamond A \supset \Box B) \supset \Box(A \supset B)}$$

  where $\mathcal{R}$ is equal to $x{\leq}y, y{\leq}z, w{\leq}u, z{\leq}t, y{\leq}t, zRw, tRu$.

- k_5:

$$\dfrac{\dfrac{\dfrac{}{x{\leq}y, yRz, z{:}\bot \Longrightarrow y{:}\bot}\,\bot^{\bullet}}{x{\leq}y, y{:}\Diamond\bot \Longrightarrow y{:}\bot}\,\Diamond^{\bullet}\ z \text{ fresh}}{\Longrightarrow x{:}\Diamond\bot \supset \bot}\,\supset^{\circ}\ y \text{ fresh}$$

Next, we have to prove all axioms of intuitionistic propositional logic can be shown in $\mathsf{labIK}_{\leq}$ as follows:

- $A \wedge B \supset B$ to the left and $A \wedge B \supset A$ to the right:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{x{\leq}y, y{\leq}y, y{:}A, y{:}B \Longrightarrow y{:}B}\,\mathsf{id_g}}{x{\leq}y, y{:}A, y{:}B \Longrightarrow y{:}B}\,{\leq}\mathsf{rf}}{x{\leq}y, y{:}A \wedge B \Longrightarrow y{:}B}\,\wedge^{\bullet}}{\Longrightarrow x{:}A \wedge B \supset B}\,\supset^{\circ}\ y \text{ fresh} \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{x{\leq}y, y{\leq}y, y{:}A, y{:}B \Longrightarrow y{:}A}\,\mathsf{id_g}}{x{\leq}y, y{:}A, y{:}B \Longrightarrow y{:}A}\,{\leq}\mathsf{rf}}{x{\leq}y, y{:}A \wedge B \Longrightarrow y{:}A}\,\wedge^{\bullet}}{\Longrightarrow x{:}A \wedge B \supset A}\,\supset^{\circ}\ y \text{ fresh}$$

- $A \supset A \vee B$ to the left and $B \supset A \vee B$ to the right:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{x{\leq}y, y{\leq}y, y{:}A \Longrightarrow y{:}A, y{:}B}\,\mathsf{id}}{x{\leq}y, y{:}A \Longrightarrow y{:}A, y{:}B}\,{\leq}\mathsf{rf}}{x{\leq}y, y{:}A \Longrightarrow y{:}A \vee B}\,\vee^{\circ}}{\Longrightarrow x{:}A \supset A \vee B}\,\supset^{\circ}\ y \text{ fresh} \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{x{\leq}y, y{\leq}y, y{:}B \Longrightarrow y{:}A, y{:}B}\,\mathsf{id}}{x{\leq}y, y{:}B \Longrightarrow y{:}A, y{:}B}\,{\leq}\mathsf{rf}}{x{\leq}y, y{:}B \Longrightarrow y{:}A \vee B}\,\vee^{\circ}}{\Longrightarrow x{:}B \supset A \vee B}\,\supset^{\circ}\ y \text{ fresh}$$

- $A \supset (B \supset (A \wedge B))$:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{x{\leq}y, y{\leq}z, y{:}A, z{:}B \Longrightarrow z{:}A}\,\mathsf{id} \quad \dfrac{\dfrac{}{x{\leq}y, y{\leq}z, z{\leq}z, y{:}A, z{:}B \Longrightarrow z{:}B}\,\mathsf{id}}{x{\leq}y, y{\leq}z, y{:}A, z{:}B \Longrightarrow z{:}B}\,{\leq}\mathsf{rf}}{x{\leq}y, y{\leq}z, y{:}A, z{:}B \Longrightarrow z{:}A \wedge B}\,\wedge^{\circ}}{x{\leq}y, y{:}A \Longrightarrow y{:}B \supset (A \wedge B)}\,z \text{ fresh}\,\supset^{\circ}}{\Longrightarrow x{:}A \supset (B \supset (A \wedge B))}\,y \text{ fresh}\,\supset^{\circ}$$

- $A \supset (B \supset A)$:

$$\dfrac{\dfrac{\dfrac{}{x{\leq}y, y{\leq}z, y{:}A, z{:}B \Longrightarrow z{:}A}\,\mathsf{id}}{x{\leq}y, y{:}A \Longrightarrow y{:}B \supset A}\,z \text{ fresh}\,\supset^{\circ}}{\Longrightarrow x{:}A \supset (B \supset A)}\,y \text{ fresh}\,\supset^{\circ}$$

- $(A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C))$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{R}, w{\leq}w, y{:}A \supset (B \supset C), z{:}A \supset B, w{:}A \Longrightarrow w{:}C, w{:}A}\,\mathsf{id}}{\mathcal{R}, y{:}A \supset (B \supset C), z{:}A \supset B, w{:}A \Longrightarrow w{:}C, w{:}A}\,{\leq}\mathsf{rf} \quad \dfrac{\mathcal{D}_1\,\|}{\mathcal{R}, w{\leq}w, z{:}A \supset B, w{:}A, w{:}B \supset C \Longrightarrow w{:}C}}{\mathcal{R}, z{:}A \supset B, w{:}A, w{:}B \supset C \Longrightarrow w{:}C}\,\supset^{\bullet}}{x{\leq}y, y{\leq}z, z{\leq}w, y{\leq}w, y{:}A \supset (B \supset C), z{:}A \supset B, w{:}A \Longrightarrow w{:}C}\,{\leq}\mathsf{tr}}{x{\leq}y, y{\leq}z, z{\leq}w, y{:}A \supset (B \supset C), z{:}A \supset B, w{:}A \Longrightarrow w{:}C}\,w \text{ fresh}\,\supset^{\circ}}{x{\leq}y, y{\leq}z, y{:}A \supset (B \supset C), z{:}A \supset B \Longrightarrow z{:}A \supset C}\,z \text{ fresh}\,\supset^{\circ}}{x{\leq}y, y{:}A \supset (B \supset C) \Longrightarrow y{:}(A \supset B) \supset (A \supset C)}\,y \text{ fresh}\,\supset^{\circ}}{\Longrightarrow x{:}(A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C))}$$

with $\mathcal{D}_1 = \dfrac{\dfrac{\dfrac{}{\mathcal{J}, z{:}A \supset B, w{:}A \Longrightarrow w{:}C, w{:}B, w{:}A}\,\mathsf{id} \quad \dfrac{}{\mathcal{J}, w{:}A, w{:}B \Longrightarrow w{:}C, w{:}B}\,\mathsf{id}}{\mathcal{R}, w{\leq}w, z{:}A \supset B, w{:}A, w{:}B \supset C \Longrightarrow w{:}C, w{:}B}\,\supset^{\bullet} \quad \dfrac{}{\mathcal{R}, w{\leq}w, z{:}A \supset B, w{:}A, w{:}C \Longrightarrow w{:}C}\,\mathsf{id}}{\mathcal{R}, w{\leq}w, z{:}A \supset B, w{:}A, w{:}B \supset C \Longrightarrow w{:}C}\,\supset^{\bullet}$

where $\mathcal{R} = x{\leq}y, y{\leq}z, z{\leq}w, y{\leq}w$ and $\mathcal{J} = \mathcal{R}, w{\leq}w, w{:}B \supset C$

- $(A \supset C) \supset ((B \supset C) \supset (A \vee B \supset C))$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{D}_1\,\| \quad \mathcal{D}_2\,\|}{\mathcal{R}, y{:}A \supset C, z{:}B \supset C, w{:}A \Longrightarrow w{:}C \quad \mathcal{R}, y{:}A \supset C, z{:}B \supset C, w{:}B \Longrightarrow w{:}C}\,\vee^{\bullet}}{x{\leq}y, y{\leq}z, z{\leq}w, y{:}A \supset C, z{:}B \supset C, w{:}A \vee B \Longrightarrow w{:}C}\,w \text{ fresh}\,\supset^{\circ}}{x{\leq}y, y{\leq}z, y{:}A \supset C, z{:}B \supset C \Longrightarrow z{:}A \vee B \supset C}\,z \text{ fresh}\,\supset^{\circ}}{x{\leq}y, y{:}A \supset C \Longrightarrow y{:}(B \supset C) \supset (A \vee B \supset C)}\,y \text{ fresh}\,\supset^{\circ}}{\Longrightarrow x{:}(A \supset C) \supset ((B \supset C) \supset (A \vee B \supset C))}$$

with $\mathcal{D}_1 = \dfrac{\dfrac{}{\mathcal{R}, y{\leq}w, y{:}A \supset C, z{:}B \supset C, w{:}A \Longrightarrow w{:}C, w{:}A}\,\mathsf{id} \quad \dfrac{}{\mathcal{R}, z{:}B \supset C, w{:}A, w{:}C \Longrightarrow w{:}C}\,\mathsf{id}}{\mathcal{R}, y{:}A \supset C, z{:}B \supset C, w{:}A \Longrightarrow w{:}C}\,\supset^{\bullet}{+}{\leq}\mathsf{tr}$

$$\text{with } \mathcal{D}_2 = \supset^\bullet \dfrac{\mathsf{id}\dfrac{}{\mathcal{R}, y{:}A \supset C, z{:}B \supset C, w{:}B \Longrightarrow w{:}C, w{:}B} \quad \mathsf{id}\dfrac{}{\mathcal{R}, y{:}A \supset C, w{:}B, w{:}C \Longrightarrow w{:}C}}{\mathcal{R}, y{:}A \supset C, z{:}B \supset C, w{:}B \Longrightarrow w{:}C}$$

where $\mathcal{R} = x{\leq}y, y{\leq}z, z{\leq}w$

- $\bot \supset A$:

$$\supset^\circ \dfrac{\bot^\bullet \dfrac{}{x{\leq}y, y{:}\bot \Longrightarrow y{:}A}}{\Longrightarrow x{:}\bot \supset A}\ y \text{ fresh}$$

Finally, we have to show how the rules of modus ponens and necessitation can be simulated in our system. For the modus ponens rule $\dfrac{A \quad A \supset B}{B}$, this is standard using the cut rule as follows:

$$\mathsf{cut}\dfrac{\mathsf{weak}\dfrac{\Longrightarrow x{:}A}{\Longrightarrow x{:}A, x{:}B} \qquad \mathsf{cut}\dfrac{\mathsf{weak}\dfrac{\Longrightarrow x{:}A \supset B}{x{:}A \Longrightarrow x{:}B, x{:}A \supset B} \qquad \leq\mathsf{rf}\dfrac{\supset^\bullet \dfrac{\mathsf{id}\dfrac{}{x{\leq}x, x{:}A, x{:}A \supset B \Longrightarrow x{:}B, x{:}A} \quad \mathsf{id}\dfrac{}{x{\leq}x, x{:}A, x{:}B \Longrightarrow x{:}B}}{x{\leq}x, x{:}A, x{:}A \supset B \Longrightarrow x{:}B}}{x{:}A, x{:}A \supset B \Longrightarrow x{:}B}}{x{:}A \Longrightarrow x{:}B}}{\Longrightarrow x{:}B}$$

We will show in the following section that the rule weak is admissible in the system $\mathsf{labIK}_{\leq}$.

For necessitation, we can transform a proof of A into a proof of $\Box A$ as follows. A proof of A is in fact a proof $\mathcal{D}$ of the sequent $\Longrightarrow z{:}A$ for some label z. If x and y are fresh labels, we can transform $\mathcal{D}$ into a proof $\mathcal{D}'$ of the sequent $x{\leq}y, yRz \Longrightarrow z{:}A$ by adding $x{\leq}y, yRz$ to every line. We can now apply the $\Box^\circ$-rule to obtain a proof of $\Longrightarrow x{:}\Box A$. $\Box$

4.4 Soundness

In order to prove the implication $3 \Longrightarrow 4$ from Theorem 4.2.2 we need to show that each sequent rule of our system $\mathsf{labIK}_{\leq}$ is sound. In other words, we want to prove that for all models $\mathcal{M}$, if $\mathcal{M}$ satisfies the premise then $\mathcal{M}$ satisfies the conclusion. To make precise what that actually means, we have to extend the relation $\Vdash$ defined in 3.1 from formulas to sequents. This is the purpose of the following two definitions:

Definition 4.4.1. Let $\mathcal{M} = \langle W, R_{\mathcal{M}}, \leq_{\mathcal{M}}, V\rangle$ be a model, and let $\mathfrak{G}$ be the sequent of the form $\mathcal{R}, \Gamma \Longrightarrow \Delta$. A $\mathfrak{G}$-*interpretation in* $\mathcal{M}$ is a mapping $[\![\cdot]\!]$ from the labels in $\mathfrak{G}$ to the set W of worlds in $\mathcal{M}$, such that whenever xRy in $\mathcal{R}$, then $[\![x]\!]R_{\mathcal{M}}[\![y]\!]$, and whenever $x{\leq}y$ in $\mathcal{R}$, then $[\![x]\!] \leq_{\mathcal{M}} [\![y]\!]$. Now we can define

$$\mathcal{M}, [\![\cdot]\!] \Vdash \mathfrak{G} \qquad \text{iff} \qquad \begin{array}{l}\text{if for all } x{:}A \in \Gamma, \text{ we have } \mathcal{M}, [\![x]\!] \Vdash A, \text{ then}\\ \text{there exists } z{:}B \in \Delta \text{ such that } \mathcal{M}, [\![z]\!] \Vdash B.\end{array} \tag{4.5}$$

Definition 4.4.2. A sequent $\mathfrak{G}$ is *satisfied* in $\mathcal{M} = \langle W, R, \leq, V\rangle$ iff for all $\mathfrak{G}$-interpretations $[\![\cdot]\!]$ we have $\mathcal{M}, [\![\cdot]\!] \Vdash \mathfrak{G}$. A sequent $\mathfrak{G}$ is *valid* in a frame $\mathcal{F} = \langle W, R, \leq\rangle$, if for all valuations V, the sequent $\mathfrak{G}$ is satisfied in $\langle W, R, \leq, V\rangle$.

We are now ready to state the main theorem of this section, of which the implication $3 \Longrightarrow 4$ in Theorem 4.2.2 is an immediate consequence.

Theorem 4.4.3. *If a sequent $\mathfrak{G}$ is provable in $\mathsf{labIK}_{\leq}$, then it is valid in every birelational frame.*

Proof. We proceed by induction on the height of the derivation of $\mathfrak{G}$, and we show for all rules r in $\mathsf{labIK}_{\leq}$

$$\mathsf{r}\,\frac{\mathfrak{G}_1 \quad \cdots \quad \mathfrak{G}_n}{\mathfrak{G}}$$

for $n \in \{0, 1, 2\}$, that whenever $\mathcal{G}_1, \ldots, \mathcal{G}_n$ are valid in all birelational frame, then so is $\mathfrak{G}$. It follows a case analysis on r:

- $\bot^{\bullet}$: This is trivial because $\bot$ is never forced.
- id: This follows immediately from Proposition 4.2.3.
- $\wedge^{\bullet}$: By way of contradiction, assume that $\mathcal{R}, \Gamma, x{:}A, x{:}B \Longrightarrow \Delta$ is valid in all bi-relational frames, but $\mathcal{R}, \Gamma, x{:}A \wedge B \Longrightarrow \Delta$ is not. This means that we have a model $\mathcal{M}$ and an interpretation $[\![\cdot]\!]$, such that $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, \Gamma, x{:}A \wedge B \Longrightarrow \Delta$, i.e., $\mathcal{M}, x \Vdash A \wedge B$. From definition of forcing, we get $\mathcal{M}, x \Vdash A$ and $\mathcal{M}, x \Vdash B$. This means we have $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, \Gamma, x{:}A, x{:}B \Longrightarrow \Delta$ and this gets a contradiction from the assumption.
- $\vee^{\circ}$: By way of contradiction, assume $\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A, x{:}B$ is valid in all bi-relational frames, but $\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \vee B$. This means that we have a model and an interpretation $[\![\cdot]\!]$, such that $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \vee B$, i.e., $\mathcal{M}, x \nVdash A \vee B$. From definition of forcing, we get $\mathcal{M}, x \nVdash A$ and $\mathcal{M}, x \nVdash B$. This means we have $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A, x{:}B$ and this get a contradiction from the assumption.
- $\Box^{\bullet}$: By way of contradiction, assume that $\mathcal{R}, x{\leq}y, yRz, \Gamma, x{:}\Box A, z{:}A \Longrightarrow \Delta$ is valid in all bi-relational frames, but $\mathcal{R}, x{\leq}y, yRz, \Gamma, x{:}\Box A \Longrightarrow \Delta$ is not. This means that we have a model $\mathcal{M}$ and an interpretation $[\![\cdot]\!]$, such that $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, x{\leq}y, yRz, \Gamma, x{:}\Box A \Longrightarrow \Delta$, i.e., $[\![x]\!] \leq_{\mathcal{M}} [\![y]\!]$ and $[\![y]\!] R_{\mathcal{M}} [\![z]\!]$ and $\mathcal{M}, x \Vdash \Box A$ and $\mathcal{M}, w \nVdash B$ for all $w{:}B \in \Delta$. However, by the definition of forcing in (3.1) we also have $\mathcal{M}, z \Vdash A$, and consequently $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, x{\leq}y, yRz, \Gamma, x{:}\Box A, z{:}A \Longrightarrow \Delta$. Then we get a contradiction.
- $\Box^{\circ}$: By way of contradiction, assume that $\mathcal{R}, x{\leq}y, yRz, \Gamma \Longrightarrow \Delta, z{:}A$ is valid in all bi-relational frames, but $\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}\Box A$ is not, where y and z do not occur in $\mathcal{R}$ or Γ or Δ. This means that we have a model $\mathcal{M}$ and an interpretation $[\![\cdot]\!]$, such that $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}\Box A$. So in particular, there are worlds y' and z' in $\mathcal{M}$ such that $[\![x]\!] \leq_{\mathcal{M}} y'$ and $y' R_{\mathcal{M}} z'$ and $\mathcal{M}, z' \nVdash A$. Now we let $[\![\cdot]\!]'$ be the extension of $[\![\cdot]\!]$ such that $[\![y]\!]' = y'$ and $[\![z]\!]' = z'$ and $[\![\cdot]\!]' = [\![\cdot]\!]$ on all other labels. Then $\mathcal{M}, [\![\cdot]\!]' \nVdash \mathcal{R}, x{\leq}y, yRz, \Gamma \Longrightarrow \Delta, z{:}A$. Then we get a contradiction.
- $\supset^{\circ}$: By way of contradiction, assume that $\mathcal{R}, x{\leq}y, \Gamma, y{:}A \Longrightarrow \Delta, y{:}B$ is valid in all bi-relational frames, but $\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \supset B$ is not, where y does not occur in $\mathcal{R}$ or Γ or Δ. This means that we have a model $\mathcal{M}$ and an interpretation $[\![\cdot]\!]$, such that $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \supset B$. So there exists a world y' in $\mathcal{M}$ such that $[\![x]\!] \leq_{\mathcal{M}} y'$

and $\mathcal{M}, y' \Vdash A$ but $\mathcal{M}, y' \nVdash B$. Now we let $[\![\cdot]\!]'$ be the extension of $[\![\cdot]\!]$ such that $[\![y]\!]' = y'$ and $[\![\cdot]\!]' = [\![\cdot]\!]$ on all other labels. Then $\mathcal{M}, [\![\cdot]\!]' \nVdash \mathcal{R}, x{\leq}y, \Gamma, y{:}A \Longrightarrow \Delta, y{:}B$. Then we get a contradiction.

- $\Diamond^\bullet$: By way of contradiction, assume that $\mathcal{R}, xRy, \Gamma, y{:}A \Longrightarrow \Delta$ is valid in all bi-relational frames, but $\mathcal{R}, \Gamma, x{:}\Diamond A \Longrightarrow \Delta$ is not, where y does not occur in $\mathcal{R}$ or Γ or Δ. This means that we have a model $\mathcal{M}$ and an interpretation $[\![\cdot]\!]$, such that $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, \Gamma, x{:}\Diamond A \Longrightarrow \Delta$, i.e. $\mathcal{M}, x \Vdash \Diamond A$. This means that there exists world y' in $\mathcal{M}$ such that $[\![x]\!] R_{\mathcal{M}} y'$ and $\mathcal{M}, y' \Vdash A$. Now we let $[\![\cdot]\!]'$ be the extension of $[\![\cdot]\!]$ such that $[\![y]\!]' = y'$ and $[\![\cdot]\!]' = [\![\cdot]\!]$ on all other labels. Then $\mathcal{M}, [\![\cdot]\!]' \nVdash \mathcal{R}, xRy, \Gamma, y{:}A \Longrightarrow \Delta$. Then we get a contradiction.

- $\Diamond^\circ$: By way of contradiction, assume that $\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A, y{:}A$ is valid in all bi-relational frames, but $\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A$ is not. This means that we have a model $\mathcal{M}$ and an interpretation $[\![\cdot]\!]$, such that $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A$, i.e., $[\![x]\!] R_{\mathcal{M}} [\![y]\!]$ and $\mathcal{M}, x \nVdash \Diamond A$. This means that for all world y' such that $[\![x]\!] R_{\mathcal{M}} y'$ we have $\mathcal{M}, y' \nVdash A$. Now we let $[\![\cdot]\!]'$ be the extension of $[\![\cdot]\!]$ such that $[\![y]\!]' = y'$. Then $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A, y{:}A$ adn we get a contradiction.

The other cases are similar (and simpler), and we leave them to the reader. In particular, note that the cases for the rules $\leq$rf, $\leq$tr, F_1 and F_2 are trivial, as all bi-relational frames have to obey the corresponding conditions. □

4.5 Cut Admissibility

Following with the proof of Theorem 4.2.2, in this section we are going to prove the admissibility of the cut rule for our system $\mathsf{labIK}_{\leq}$.

Theorem 4.5.1. *The* cut *rule is admissible for* $\mathsf{labIK}_{\leq}$.

This theorem directly entails the implication $2 \Longrightarrow 3$ of Theorem 4.2.2. But before we can prove it, we need a series of auxiliary lemmas.

The *height* of a derivation $\mathcal{D}$, denoted by $|\mathcal{D}|$, is the height of $\mathcal{D}$ when seen as a tree, i.e., the length of the longest path in the tree from its root to one of its leaves. We say that a rule is *height-preserving admissible* if for every derivation $\mathcal{D}$ of its premise(s) there is a derivation $\mathcal{D}'$ of its conclusion such that $|\mathcal{D}'| \leq |\mathcal{D}|$. A rule is *height-preserving invertible* if for every derivation of the conclusion of the rule there are derivations for each of its premises with at most the same height.

The first lemma is the height-preserving admissibility of weakening on both relational atoms and labelled formulas.

Lemma 4.5.2. *The weakening rule* $\mathsf{weak}\dfrac{\mathcal{R}, \Gamma \Longrightarrow \Delta}{\mathcal{R}, \mathcal{R}', \Gamma, \Gamma' \Longrightarrow \Delta, \Delta'}$ *is height-preserving admissible for* $\mathsf{labIK}_{\leq}$.

Proof. By a straightforward induction on the height of the derivation, we can transform any derivation

$$\begin{array}{c} \mathcal{D} \\ \| \\ \mathcal{R}, \Gamma \Longrightarrow \Delta \end{array} \quad \text{into} \quad \begin{array}{c} \mathcal{D}^{\mathsf{w}} \\ \| \\ \mathcal{R}, \mathcal{R}', \Gamma, \Gamma' \Longrightarrow \Delta, \Delta' \end{array}$$

of the same (or smaller) height. □

The next lemma looks like a special case of Proposition 4.2.3, but it is not. First, we need to preserve the height, and second, we cannot prove it using cut rule as we are trying to eliminate it from derivations.

Lemma 4.5.3. *The atomic version of* $\mathsf{mon}^\bullet$ *and* mon°

$$\mathsf{mon}_\mathsf{a}^\bullet \frac{\mathcal{R}, x{\leq}y, \Gamma, x{:}a, y{:}a \Longrightarrow \Delta}{\mathcal{R}, x{\leq}y, \Gamma, x{:}a \Longrightarrow \Delta} \qquad \mathsf{mon}_\mathsf{a}^\circ \frac{\mathcal{R}, x{\leq}y, \Gamma \Longrightarrow \Delta, x{:}a, y{:}a}{\mathcal{R}, x{\leq}y, \Gamma \Longrightarrow \Delta, y{:}a}$$

are height-preserving admissible for $\mathsf{labIK}_\leq$.

Proof. By induction on the height of $\mathcal{D}$, we prove that for any proof of $\mathcal{R}, x{\leq}x', \Gamma, x{:}a, x'{:}a \Longrightarrow \Delta$, there exists a proof of $\mathcal{R}, x{\leq}x', \Gamma, x{:}a \Longrightarrow \Delta$ of the same (or smaller) height. The inductive step is straightforward by permutation of rules. The base cases are obtained as follows:

$$\mathsf{mon}_\mathsf{a}^\bullet \frac{\mathsf{id}\dfrac{}{\mathcal{R}, x{\leq}x', x'{\leq}x'', \Gamma, x{:}a, x'{:}a \Longrightarrow \Delta, x''{:}a}}{\mathcal{R}, x{\leq}x', x'{\leq}x'', \Gamma, x{:}a \Longrightarrow \Delta, x''{:}a} \quad \rightsquigarrow \quad \leq\mathsf{tr} \frac{\mathsf{id}\dfrac{}{\mathcal{R}, x{\leq}x', x'{\leq}x'', x{\leq}x'', \Gamma, x{:}a \Longrightarrow \Delta, x''{:}a}}{\mathcal{R}, x{\leq}x', x'{\leq}x'', \Gamma, x{:}a \Longrightarrow \Delta, x''{:}a}$$

$$\mathsf{mon}_\mathsf{a}^\bullet \frac{\mathsf{id}\dfrac{}{\mathcal{R}, x{\leq}x', x{:}a, x'{:}a \Longrightarrow \Delta, x{:}a}}{\mathcal{R}, x{\leq}x', \Gamma, x{:}a \Longrightarrow \Delta, x{:}a} \quad \rightsquigarrow \quad \leq\mathsf{rf} \frac{\mathsf{id}\dfrac{}{\mathcal{R}, x{\leq}x', x{\leq}x, x{:}a \Longrightarrow \Delta, x{:}a}}{\mathcal{R}, x{\leq}x', \Gamma, x{:}a \Longrightarrow \Delta, x{:}a}$$

$$\mathsf{mon}_\mathsf{a}^\bullet \frac{\mathsf{id}\dfrac{}{\mathcal{R}, x{\leq}x', x{:}a, x'{:}a \Longrightarrow \Delta, x'{:}a}}{\mathcal{R}, x{\leq}x', \Gamma, x{:}a \Longrightarrow \Delta, x'{:}a} \quad \rightsquigarrow \quad \mathsf{id}\frac{}{\mathcal{R}, x{\leq}x', \Gamma, x{:}a \Longrightarrow \Delta, x'{:}a}$$

□

The next lemma shows that all the rules in our system are invertible, as already mentioned in the motivation in Section 4.1.

Lemma 4.5.4. *All single-premise rules of* $\mathsf{labIK}_\leq$ *are height-preserving invertible. Furthermore, the rules* $\vee^\bullet$, $\wedge^\circ$ *and* $\supset^\bullet$ *are height-preserving invertible on both premises.*

Proof. For each rule r, we need to show that if there exists a proof $\mathcal{D}$ of the conclusion, there exists a proof $\mathcal{D}^{\mathsf{r}_i}$ of the i-th premise, of the same (or smaller) height. For $\wedge^\circ$, $\wedge^\bullet$, $\vee^\circ$, $\vee^\bullet$ and $\supset^\bullet$ we use a standard induction on the height of $\mathcal{D}$. For $\supset^\circ$, $\Box^\circ$, $\Diamond^\bullet$ as well, but we need to make sure that the obtained derivation uses a fresh label by using substitution inside $\mathcal{D}^{\mathsf{r}}$ when necessary. The other rules can be shown invertible using Lemma 4.5.2. □

The next lemma is the central ingredient of our cut elimination proof.

Lemma 4.5.5. *Given a derivation of shape*

$$\mathsf{cut} \frac{\mathcal{R}, \Gamma \Longrightarrow \Delta, z{:}C \;\; (\mathcal{D}_1) \qquad \mathcal{R}, \Gamma, z{:}C \Longrightarrow \Delta \;\; (\mathcal{D}_2)}{\mathcal{R}, \Gamma \Longrightarrow \Delta}$$

where $\mathcal{D}_1$ *and* $\mathcal{D}_2$ *are both cut-free, there is a cut-free derivation of* $\mathcal{R}, \Gamma \Longrightarrow \Delta$

Proof. The proof is by a lexicographic induction on the complexity of the cut-formula C and the sum of the heights $|\mathcal{D}_1| + |\mathcal{D}_2|$. We perform a case analysis on the last rule used in $\mathcal{D}_1$ above the cut and whether it applies to the cut-formula or not. In case it does not, we are in a *commutative* case; in case it does, we have to perform a similar analysis on $\mathcal{D}_2$ to end up in a *key* case.

Base cases: When the last rule in $\mathcal{D}_1$ is an axiom, we can produce directly a cut-free derivation of the conclusion. In the first case, we appeal to Lemma 4.5.3, to use the atomic monotonicity rule freely and to Lemma 4.5.2 to obtain $\mathcal{D}_2^{\mathsf{w}}$ through weakening admissibility.

- $$\mathsf{cut}\,\dfrac{\mathsf{id}\,\dfrac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}a \Longrightarrow \Delta, y{:}a} \quad \begin{array}{c}\mathcal{D}_2\\ \Vert\\ \mathcal{R}, x{\leq}y, \Gamma, x{:}a, y{:}a \Longrightarrow \Delta\end{array}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}a \Longrightarrow \Delta} \quad \rightsquigarrow \quad \mathsf{mon_a}^{\bullet}\,\dfrac{\begin{array}{c}\mathcal{D}_2^{\mathsf{w}}\\ \Vert\\ \mathcal{R}, x{\leq}y, \Gamma, x{:}a, y{:}a \Longrightarrow \Delta\end{array}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}a \Longrightarrow \Delta}$$
- $$\mathsf{cut}\,\dfrac{\mathsf{id}\,\dfrac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}a \Longrightarrow \Delta, y{:}a, z{:}C} \quad \begin{array}{c}\mathcal{D}_2\\ \Vert\\ \mathcal{R}, x{\leq}y, \Gamma, x{:}a, z{:}C \Longrightarrow \Delta, y{:}a\end{array}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}a \Longrightarrow \Delta, y{:}a} \quad \rightsquigarrow \quad \mathsf{id}\,\dfrac{}{\mathcal{R}, x{\leq}y, \Gamma, x{:}a \Longrightarrow \Delta, y{:}a}$$
- $$\mathsf{cut}\,\dfrac{\bot^{\bullet}\,\dfrac{}{\mathcal{R}, \Gamma, x{:}\bot \Longrightarrow \Delta, z{:}C} \quad \begin{array}{c}\mathcal{D}_2\\ \Vert\\ \mathcal{R}, \Gamma, x{:}\bot, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, \Gamma, x{:}\bot \Longrightarrow \Delta} \quad \rightsquigarrow \quad \bot^{\bullet}\,\dfrac{}{\mathcal{R}, \Gamma, x{:}\bot \Longrightarrow \Delta}$$

Commutative cases: In such a case, the complexity of the cut-formula stays constant, but the height of the derivation above the cut decreases.

- $\supset^{\bullet}$:

$$\supset^{\bullet}\,\dfrac{\mathsf{cut}\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \Vert\\ \mathcal{R}, x{\leq}y, \Gamma, x{:}A \supset B \Longrightarrow \Delta, z{:}C, y{:}A\end{array} \quad \begin{array}{c}\mathcal{D}_2\\ \Vert\\ \mathcal{R}, x{\leq}y, \Gamma, y{:}B \Longrightarrow \Delta, z{:}C\end{array}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \supset B \Longrightarrow \Delta, z{:}C} \quad \begin{array}{c}\mathcal{D}_3\\ \Vert\\ \mathcal{R}, x{\leq}y, \Gamma, x{:}A \supset B, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \supset B \Longrightarrow \Delta}$$

$$\rightsquigarrow \quad \supset^{\bullet}\,\dfrac{\begin{array}{c}\mathcal{D}_1'\\ \Vert\\ \mathcal{R}, x{\leq}y, \Gamma, x{:}A \supset B \Longrightarrow \Delta, y{:}A\end{array} \quad \mathsf{cut}\,\dfrac{\begin{array}{c}\mathcal{D}_2\\ \Vert\\ \mathcal{R}, x{\leq}y, \Gamma, y{:}B \Longrightarrow \Delta, z{:}C\end{array} \quad \begin{array}{c}\mathcal{D}_3^{\supset\bullet\prime}\\ \Vert\\ \mathcal{R}, x{\leq}y, \Gamma, y{:}B, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, x{\leq}y, \Gamma, y{:}B \Longrightarrow \Delta}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \supset B \Longrightarrow \Delta}$$

with $\mathcal{D}_1' = \mathsf{cut}\dfrac{\overset{\mathcal{D}_1}{\Vert} \quad \overset{\mathcal{D}_3^{\supset\bullet\prime\prime}}{\Vert}}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \supset B \Longrightarrow \Delta, z{:}C, y{:}A \quad \mathcal{R}, x{\leq}y, \Gamma, x{:}A \supset B, z{:}C \Longrightarrow \Delta, y{:}A}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \supset B \Longrightarrow \Delta, y{:}A}$

where $\mathcal{D}_3^{\supset\bullet\prime}$ and $\mathcal{D}_3^{\supset\bullet\prime\prime}$ are obtained using Lemma 4.5.4 for the rule $\supset^\bullet$ which is invertible in the right premise and the left premise, respectively.

- $\supset^\circ$:

$$\supset^\circ \dfrac{\mathsf{cut}\dfrac{\overset{\mathcal{D}_1}{\Vert}}{\mathcal{R}, x{\leq}x', \Gamma, x'{:}A \Longrightarrow \Delta, x'{:}B, z{:}C}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \supset B, z{:}C} \; x' \text{ fresh} \quad \overset{\mathcal{D}_2}{\Vert}}{\mathcal{R}, \Gamma, z{:}C \Longrightarrow \Delta, x{:}A \supset B}}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \supset B}$$

$$\rightsquigarrow \quad \mathsf{cut}\dfrac{\overset{\mathcal{D}_1[x''/x']}{\Vert} \quad \overset{\mathcal{D}_2^{\supset\circ}}{\Vert}}{\mathcal{R}, x{\leq}x'', \Gamma, x''{:}A \Longrightarrow \Delta, x''{:}B, z{:}C \quad \mathcal{R}, x{\leq}x'', \Gamma, z{:}C, x''{:}A \Longrightarrow \Delta, x''{:}B}}{\supset^\circ \dfrac{\mathcal{R}, x{\leq}x'', \Gamma, x''{:}A \Longrightarrow \Delta, x''{:}B}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \supset B} \; x'' \text{ fresh (also in } \mathcal{D}_2)}$$

where $\mathcal{D}_2^{\supset\circ}$ is obtained using Lemma 4.5.4. We use the same naming scheme in the following cases.

- $\Box^\bullet$:

$$\Box^\bullet \dfrac{\mathsf{cut}\dfrac{\overset{\mathcal{D}_1}{\Vert}}{\mathcal{R}, x{\leq}u, uRv, \Gamma, x{:}\Box A, v{:}A \Longrightarrow \Delta, z{:}C}{\mathcal{R}, x{\leq}u, uRv, \Gamma, x{:}\Box \Longrightarrow \Delta, z{:}C} \quad \overset{\mathcal{D}_2}{\Vert}}{\mathcal{R}, x{\leq}u, uRv, \Gamma, x{:}\Box A, z{:}C \Longrightarrow \Delta}}{\mathcal{R}, x{\leq}u, uRv, \Gamma, x{:}\Box A \Longrightarrow \Delta}$$

$$\rightsquigarrow \quad \Box^\bullet \dfrac{\mathsf{cut}\dfrac{\overset{\mathcal{D}_1}{\Vert} \quad \overset{\mathcal{D}_2^{\mathsf{w}}}{\Vert}}{\mathcal{R}, x{\leq}u, uRv, \Gamma, x{:}\Box A, v{:}A \Longrightarrow \Delta, z{:}C \quad \mathcal{R}, x{\leq}u, uRv, \Gamma, x{:}\Box A, v{:}A, z{:}C \Longrightarrow \Delta}}{\mathcal{R}, x{\leq}u, uRv, \Gamma, x{:}\Box A, v{:}A \Longrightarrow \Delta}}{\mathcal{R}, x{\leq}u, uRv, \Gamma, x{:}\Box A \Longrightarrow \Delta}$$

where $\mathcal{D}_2^{\mathsf{w}}$ is obtained using Lemma 4.5.2. We use the same naming scheme in the following cases.

- $\Box^\circ$:

$$\Box^\circ \dfrac{\mathsf{cut}\dfrac{\overset{\mathcal{D}_1}{\Vert}}{\mathcal{R}, x{\leq}x', x'Ry', \Gamma \Longrightarrow \Delta, y'{:}A, z{:}C}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}\Box A, z{:}C} \; x', y' \text{ fresh} \quad \overset{\mathcal{D}_2}{\Vert}}{\mathcal{R}, \Gamma, z{:}C \Longrightarrow \Delta, x{:}\Box A}}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}\Box A}$$

$$\rightsquigarrow \quad \mathsf{cut}\dfrac{\overset{\mathcal{D}_1[u/x'][v/y']}{\Vert} \quad \overset{\mathcal{D}_2^{\Box\circ}}{\Vert}}{\mathcal{R}, x{\leq}u, uRv, \Gamma \Longrightarrow \Delta, v{:}A, z{:}C \quad \mathcal{R}, x{\leq}u, uRv, \Gamma, z{:}C \Longrightarrow \Delta, v{:}A}}{\Box^\circ \dfrac{\mathcal{R}, x{\leq}u, uRv, \Gamma \Longrightarrow \Delta, v{:}A}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}\Box A} \; u, v \text{ fresh (also in } \mathcal{D}_2)}$$

- $\Diamond^\bullet$:

$$\mathsf{cut}\,\dfrac{\Diamond^\bullet\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \mathcal{R}, xRy', \Gamma, y'{:}A \Longrightarrow \Delta, z{:}C\end{array}}{\mathcal{R}, \Gamma, x{:}\Diamond A \Longrightarrow \Delta, z{:}C}\, y' \text{ fresh} \qquad \begin{array}{c}\mathcal{D}_2\\ \mathcal{R}, \Gamma, x{:}\Diamond A, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, \Gamma, x{:}\Diamond A \Longrightarrow \Delta}$$

$$\rightsquigarrow \quad \Diamond^\bullet\,\dfrac{\mathsf{cut}\,\dfrac{\begin{array}{c}\mathcal{D}_1[y''/y']\\ \mathcal{R}, xRy'', \Gamma, y''{:}A \Longrightarrow \Delta, z{:}C\end{array} \quad \begin{array}{c}\mathcal{D}_2^{\Diamond^\bullet}\\ \mathcal{R}, xRy'', \Gamma, y''{:}A, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, xRy'', \Gamma, y''{:}A \Longrightarrow \Delta}}{\mathcal{R}, \Gamma, x{:}\Diamond A \Longrightarrow \Delta}\, y'' \text{ fresh (also in } \mathcal{D}_2)$$

- $\Diamond^\circ$:

$$\mathsf{cut}\,\dfrac{\Diamond^\circ\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A, y{:}A, z{:}C\end{array}}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A, z{:}C} \qquad \begin{array}{c}\mathcal{D}_2\\ \mathcal{R}, \Gamma, z{:}C \Longrightarrow \Delta, x{:}\Diamond A\end{array}}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A}$$

$$\rightsquigarrow \quad \Diamond^\circ\,\dfrac{\mathsf{cut}\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A, y{:}A, z{:}C\end{array} \quad \begin{array}{c}\mathcal{D}_2^{\mathsf{w}}\\ \mathcal{R}, \Gamma, z{:}C \Longrightarrow \Delta, x{:}\Diamond A, y{:}A\end{array}}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A, y{:}A}}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A}$$

- $\leq\mathsf{rf}$:

$$\mathsf{cut}\,\dfrac{\leq\mathsf{rf}\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \mathcal{R}, x{\leq}x, \Gamma \Longrightarrow \Delta, z{:}C\end{array}}{\mathcal{R}, \Gamma \Longrightarrow \Delta, z{:}C} \qquad \begin{array}{c}\mathcal{D}_2\\ \mathcal{R}, \Gamma, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, \Gamma \Longrightarrow \Delta}$$

$$\rightsquigarrow \quad \leq\mathsf{rf}\,\dfrac{\mathsf{cut}\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \mathcal{R}, x{\leq}x, \Gamma \Longrightarrow \Delta, z{:}C\end{array} \quad \begin{array}{c}\mathcal{D}_2^{\mathsf{w}}\\ \mathcal{R}, x{\leq}x, \Gamma, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, x{\leq}x, \Gamma \Longrightarrow \Delta}}{\mathcal{R}, \Gamma \Longrightarrow \Delta}$$

- $\leq\mathsf{tr}$:

$$\mathsf{cut}\,\dfrac{\leq\mathsf{tr}\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \mathcal{R}, x{\leq}y, y{\leq}z, x{\leq}z, \Gamma \Longrightarrow \Delta, z{:}C\end{array}}{\mathcal{R}, x{\leq}y, y{\leq}z, \Gamma \Longrightarrow \Delta, z{:}C} \qquad \begin{array}{c}\mathcal{D}_2\\ \mathcal{R}, x{\leq}y, y{\leq}z, \Gamma, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, x{\leq}y, y{\leq}z, \Gamma \Longrightarrow \Delta}$$

$$\rightsquigarrow \quad \leq\mathsf{tr}\,\dfrac{\mathsf{cut}\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \mathcal{R}, x{\leq}y, y{\leq}z, x{\leq}z, \Gamma \Longrightarrow \Delta, z{:}C\end{array} \quad \begin{array}{c}\mathcal{D}_2^{\mathsf{w}}\\ \mathcal{R}, x{\leq}y, y{\leq}z, x{\leq}z, \Gamma, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, x{\leq}y, y{\leq}z, x{\leq}z, \Gamma \Longrightarrow \Delta}}{\mathcal{R}, x{\leq}y, y{\leq}z, \Gamma \Longrightarrow \Delta}$$

- F_1:

$$\mathsf{cut}\,\dfrac{\mathsf{F}_1\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \|\\ \mathcal{R}, xRy, y{\leq}z, x{\leq}u, uRz, \Gamma \Longrightarrow \Delta, z{:}C\end{array}}{\mathcal{R}, xRy, y{\leq}z, \Gamma \Longrightarrow \Delta, z{:}C}\, u \text{ fresh} \qquad \begin{array}{c}\mathcal{D}_2\\ \|\\ \mathcal{R}, xRy, y{\leq}z, \Gamma, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, xRy, y{\leq}z, \Gamma \Longrightarrow \Delta}$$

$$\rightsquigarrow \quad \mathsf{F}_1\,\dfrac{\mathsf{cut}\,\dfrac{\begin{array}{c}\mathcal{D}_1[v/u]\\ \|\\ \mathcal{R}, xRy, y{\leq}z, x{\leq}v, vRz, \Gamma \Longrightarrow \Delta, z{:}C\end{array} \quad \begin{array}{c}\mathcal{D}_2^{\mathsf{w}}\\ \|\\ \mathcal{R}, xRy, y{\leq}z, x{\leq}v, vRz, \Gamma, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, xRy, y{\leq}z, x{\leq}v, vRz, \Gamma \Longrightarrow \Delta}}{\mathcal{R}, xRy, y{\leq}z, \Gamma \Longrightarrow \Delta}\, v \text{ fresh (also in } \mathcal{D}_2)$$

- F_2:

$$\mathsf{cut}\,\dfrac{\mathsf{F}_2\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \|\\ \mathcal{R}, xRy, x{\leq}z, y{\leq}u, zRu, \Gamma \Longrightarrow \Delta, z{:}C\end{array}}{\mathcal{R}, xRy, x{\leq}z, \Gamma \Longrightarrow \Delta, z{:}C}\, u \text{ fresh} \qquad \begin{array}{c}\mathcal{D}_2\\ \|\\ \mathcal{R}, xRy, x{\leq}z, \Gamma, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, xRy, x{\leq}z, \Gamma \Longrightarrow \Delta}$$

$$\rightsquigarrow \quad \mathsf{F}_2\,\dfrac{\mathsf{cut}\,\dfrac{\begin{array}{c}\mathcal{D}_1[v/u]\\ \|\\ \mathcal{R}, xRy, x{\leq}z, y{\leq}v, zRv, \Gamma \Longrightarrow \Delta, z{:}C\end{array} \quad \begin{array}{c}\mathcal{D}_2^{\mathsf{w}}\\ \|\\ \mathcal{R}, xRy, x{\leq}z, y{\leq}v, zRv, \Gamma, z{:}C \Longrightarrow \Delta\end{array}}{\mathcal{R}, xRy, x{\leq}z, y{\leq}v, zRv, \Gamma \Longrightarrow \Delta}}{\mathcal{R}, xRy, x{\leq}z, \Gamma \Longrightarrow \Delta}\, v \text{ fresh (also in } \mathcal{D}_2)$$

Key cases: If the last rule in $\mathcal{D}_1$ and the last rule in $\mathcal{D}_2$ both apply to the cut-formulas, then it is the complexity of the cut-formula that is the decreasing inductive measure, save for the modal cases, where it is important to note the combination of induction on both height and formula size.

- $C = A \wedge B$:

$$\mathsf{cut}\,\dfrac{\wedge^{\circ}\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \|\\ \mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A\end{array} \quad \begin{array}{c}\mathcal{D}_2\\ \|\\ \mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}B\end{array}}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \wedge B} \quad \wedge^{\bullet}\,\dfrac{\begin{array}{c}\mathcal{D}_3\\ \|\\ \mathcal{R}, \Gamma, x{:}A, x{:}B \Longrightarrow \Delta\end{array}}{\mathcal{R}, \Gamma, x{:}A \wedge B \Longrightarrow \Delta}}{\mathcal{R}, \Gamma \Longrightarrow \Delta}$$

$$\rightsquigarrow \quad \mathsf{cut}\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \|\\ \mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A\end{array} \quad \mathsf{cut}\,\dfrac{\begin{array}{c}\mathcal{D}_2^{\mathsf{w}}\\ \|\\ \mathcal{R}, \Gamma, x{:}A \Longrightarrow \Delta, x{:}B\end{array} \quad \begin{array}{c}\mathcal{D}_3\\ \|\\ \mathcal{R}, \Gamma, x{:}A, x{:}B \Longrightarrow \Delta\end{array}}{\mathcal{R}, \Gamma, x{:}A \Longrightarrow \Delta}}{\mathcal{R}, \Gamma \Longrightarrow \Delta}$$

- $C = A \vee B$:

$$\mathsf{cut}\,\dfrac{\vee^{\circ}\,\dfrac{\begin{array}{c}\mathcal{D}_1\\ \|\\ \mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A, x{:}B\end{array}}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A \vee B} \quad \vee^{\bullet}\,\dfrac{\begin{array}{c}\mathcal{D}_2\\ \|\\ \mathcal{R}, \Gamma, x{:}A \Longrightarrow \Delta\end{array} \quad \begin{array}{c}\mathcal{D}_3\\ \|\\ \mathcal{R}, \Gamma, x{:}B \Longrightarrow \Delta\end{array}}{\mathcal{R}, \Gamma, x{:}A \vee B \Longrightarrow \Delta}}{\mathcal{R}, \Gamma \Longrightarrow \Delta}$$

$$\rightsquigarrow \quad \mathsf{cut}\,\dfrac{\mathsf{cut}\,\dfrac{\mathcal{D}_1\Big\|}{\mathcal{R},\Gamma\Longrightarrow\Delta,x{:}A,x{:}B}\quad\dfrac{\mathcal{D}_3^{\mathsf{w}}\Big\|}{\mathcal{R},\Gamma,x{:}B\Longrightarrow\Delta,x{:}A}}{\mathcal{R},\Gamma\Longrightarrow\Delta,x{:}A}\qquad\dfrac{\mathcal{D}_2\Big\|}{\mathcal{R},\Gamma,x{:}A\Longrightarrow\Delta}}{\mathcal{R},\Gamma\Longrightarrow\Delta}$$

- $C = A \supset B$:

$$\mathsf{cut}\,\dfrac{\supset^{\circ}\,\dfrac{\mathcal{D}_1\Big\|}{\mathcal{R},x{\leq}y,x{\leq}x',\Gamma,x'{:}A\Longrightarrow\Delta,x'{:}B}}{\mathcal{R},x{\leq}y,\Gamma\Longrightarrow\Delta,x{:}A\supset B}\quad\dfrac{\supset^{\bullet}\,\dfrac{\mathcal{D}_2\Big\|}{\mathcal{R},x{\leq}y,\Gamma,x{:}A\supset B\Longrightarrow\Delta,y{:}A}\quad\dfrac{\mathcal{D}_3\Big\|}{\mathcal{R},x{\leq}y,\Gamma,y{:}B\Longrightarrow\Delta}}{\mathcal{R},x{\leq}y,\Gamma,x{:}A\supset B\Longrightarrow\Delta}}{\mathcal{R},x{\leq}y,\Gamma\Longrightarrow\Delta}$$

$$\rightsquigarrow \quad \mathsf{cut}\,\dfrac{\dfrac{\mathcal{D}_1'\Big\|}{\mathcal{R},x{\leq}y,\Gamma\Longrightarrow\Delta,y{:}A}\quad\mathsf{cut}\,\dfrac{\dfrac{\mathcal{D}_1[y/x']\Big\|}{\mathcal{R},x{\leq}y,\Gamma,y{:}A\Longrightarrow\Delta,y{:}B}\quad\dfrac{\mathcal{D}_3\Big\|}{\mathcal{R},x{\leq}y,\Gamma,y{:}B\Longrightarrow\Delta}}{\mathcal{R},x{\leq}y,\Gamma,y{:}A\Longrightarrow\Delta}}{\mathcal{R},x{\leq}y,\Gamma\Longrightarrow\Delta}$$

$$\text{with}\quad \mathcal{D}_1' = \supset^{\circ}\,\dfrac{\dfrac{\mathcal{D}_1^{\mathsf{w}}\Big\|}{\mathcal{R},x{\leq}y,x{\leq}x',\Gamma,x'{:}A\Longrightarrow\Delta,x'{:}B,y{:}A}}{\mathcal{R},x{\leq}y,\Gamma\Longrightarrow\Delta,x{:}A\supset B,y{:}A}\quad\dfrac{\mathcal{D}_2\Big\|}{\mathcal{R},x{\leq}y,\Gamma,x{:}A\supset B\Longrightarrow\Delta,y{:}A}}{\mathcal{R},x{\leq}y,\Gamma\Longrightarrow\Delta,y{:}A}$$

- $C = \Box A$:

$$\mathsf{cut}\,\dfrac{\Box^{\circ}\,\dfrac{\mathcal{D}_1\Big\|}{\mathcal{R},x{\leq}u,uRv,x{\leq}x',x'Ry',\Gamma\Longrightarrow\Delta,y'{:}A}}{\mathcal{R},x{\leq}u,uRv,\Gamma\Longrightarrow\Delta,x{:}\Box A}\quad\dfrac{\Box^{\bullet}\,\dfrac{\mathcal{D}_2\Big\|}{\mathcal{R},x{\leq}u,uRv,\Gamma,x{:}\Box A,v{:}A\Longrightarrow\Delta}}{\mathcal{R},x{\leq}u,uRv,\Gamma,x{:}\Box A\Longrightarrow\Delta}}{\mathcal{R},x{\leq}u,uRv,\Gamma\Longrightarrow\Delta}$$

$$\rightsquigarrow \quad \mathsf{cut}\,\dfrac{\dfrac{\mathcal{D}_1[u/x',v/y']\Big\|}{\mathcal{R},x{\leq}u,uRv,\Gamma\Longrightarrow\Delta,v{:}A}\quad\dfrac{\mathcal{D}_2'\Big\|}{\mathcal{R},x{\leq}u,uRv,\Gamma,v{:}A\Longrightarrow\Delta}}{\mathcal{R},x{\leq}u,uRv,\Gamma\Longrightarrow\Delta}$$

$$\text{with}\quad \mathcal{D}_2' = \Box^{\circ}\,\dfrac{\dfrac{\mathcal{D}_1^{\mathsf{w}}\Big\|}{\mathcal{R},x{\leq}u,uRv,x{\leq}x',x'Ry',\Gamma,v{:}A\Longrightarrow\Delta,x{:}\Box A,y'{:}A}}{\mathcal{R},x{\leq}u,uRv,\Gamma,v{:}A\Longrightarrow\Delta,x{:}\Box A}\quad\dfrac{\mathcal{D}_2\Big\|}{\mathcal{R},x{\leq}u,uRv,\Gamma,x{:}\Box A,v{:}A\Longrightarrow\Delta}}{\mathcal{R},x{\leq}u,uRv,\Gamma,v{:}A\Longrightarrow\Delta}$$

The top cut is admissible by induction on the height, as the size of the cut-formula is constant. This however may increase the height above the right premise of the bottom cut arbitrarily. The bottom cut is still admissible as the size of the cut-formula decreases.

- $C = \Diamond A$:

$$\mathsf{cut}\,\dfrac{\Diamond^{\circ}\,\dfrac{\mathcal{D}_1\Big\|}{\mathcal{R},xRy,\Gamma\Longrightarrow\Delta,x{:}\Diamond A,y{:}A}}{\mathcal{R},xRy,\Gamma\Longrightarrow\Delta,x{:}\Diamond A}\quad\dfrac{\Diamond^{\bullet}\,\dfrac{\mathcal{D}_2\Big\|}{\mathcal{R},xRy,xRy',\Gamma,y'{:}A\Longrightarrow\Delta}}{\mathcal{R},xRy,\Gamma,x{:}\Diamond A\Longrightarrow\Delta}\;y' \text{ is fresh}}{\mathcal{R},xRy,\Gamma\Longrightarrow\Delta}$$

$$\rightsquigarrow \quad \mathsf{cut}\,\dfrac{\dfrac{\mathcal{D}_1\Vert}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, y{:}A, x{:}\Diamond A} \quad \Diamond^{\bullet}\dfrac{\dfrac{\mathcal{D}_2^{\mathsf{w}}\Vert}{\mathcal{R}, xRy, xRy', \Gamma, y'{:}A \Longrightarrow \Delta, y{:}A}\, y' \text{ is fresh}}{\mathcal{R}, xRy, \Gamma, x{:}\Diamond A \Longrightarrow \Delta, y{:}A}}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, y{:}A} \quad \mathsf{cut}\,\dfrac{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, y{:}A \qquad \mathcal{D}_2[y/y']\Vert \; \mathcal{R}, xRy, \Gamma, y{:}A \Longrightarrow \Delta}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta}$$

The induction hypothesis is applied here again twice as above, on the height for the top cut and on formula size for the bottom one. □

We can now complete the proof of Theorem 4.5.1.

Proof of Theorem 4.5.1. By induction on the number of cut rules in $\mathcal{D}$, always applying Lemma 4.5.5 to the leftmost topmost cut. □

4.6 Extensions

The main goal of this section is to generate stronger logics adding new axioms to our system. We say *stronger logic* to refer to the fact that we are restricting the class of frames we want to consider, imposing some restrictions on the accessibility relation R. We will present extensions for the fully labelled sequent system $\mathsf{labIK}_{\leq}$ with *one-sided Scott-Lemmon axioms* and with *path Scott-Lemmon axioms*.

4.6.1 One-sided Scott-Lemmon axioms

In the *fully labelled* framework, we are also able to consider the logics defined by *one-sided intuitionistic Scott-Lemmon axioms* introduced also in Section 3.2 as follows:

$$\Diamond^k \Box^l A \supset \Box^m \Diamond^n A \tag{4.6}$$

for any natural numbers k, l, m, n. These axioms satisfy the *intuitionistic klmn-incestuality condition* presented in Chapter 3 in Figure 3.1.

Following again the axiom-as-rule idea, to have a sound and complete system for IK extended by *one-sided intuitionistic Scott-Lemmon axioms*, we introduce to the system $\mathsf{labIK}_{\leq}$ the $\mathsf{g}_{\mathsf{klmn}}$ rule, for any natural numbers k, l, m, n:

$$\mathsf{g}_{\mathsf{klmn}}\,\frac{\mathcal{R}, xR^k y, xR^m z, y{\leq}y', y'R^l u, zR^n u, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xR^k y, xR^m z, \Gamma \Longrightarrow \Delta}\, y', u \text{ fresh} \tag{4.7}$$

where $xR^k y$ is an abbreviation for $xRx_1, x_1Rx_2, \ldots, x_{k-1}Ry$, and the labels $x, x_1, \ldots, x_{k-1}, y$ do *not* have to be distinct, and if $k = 0$ then $x = y$; and similarly for $xR^m z$, if $m = 0$, then $x = z$. In case some of them coincide, they are not repeated since $\mathcal{R}$ is a *set* of relational atoms. However, in the premise, the expression $y'R^l u$ stands for $y'Ru_1, \ldots, u_{l-1}Ru$ where all $u_1, \ldots, u_{l-1}, u$ are fresh, and therefore pairwise distinct, except if $l = 0$, in which case $u = y'$; similarly for $zR^n u$, but note that if $n = 0$ then $u = z$.

For example, the derivation in (4.8) below requires the (valid) application of the rule

$$\mathsf{g}_{1110}\,\frac{\mathcal{R}, xRy, xRz, y{\leq}y', y'Rz, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xRy, xRz, \Gamma \Longrightarrow \Delta}\, y' \text{ fresh} \qquad \text{as} \qquad \mathsf{g}_{1110}\,\frac{\mathcal{R}, xRz, z{\leq}y', y'Rz, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xRz, \Gamma \Longrightarrow \Delta}\, y' \text{ fresh}$$

i.e., the case where $y = z$, in order to derive $\Box(\Box A \supset A)$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x{\leq}y, yRz, z{\leq}v, y{\leq}u, uRv, v{\leq}w, wRv, v{:}\Box A, v{:}A \Longrightarrow v{:}A}\,\mathsf{id}}{x{\leq}y, yRz, z{\leq}v, y{\leq}u, uRv, v{\leq}w, wRv, v{:}\Box A \Longrightarrow v{:}A}\,\Box^{\bullet}}{x{\leq}y, yRz, z{\leq}v, y{\leq}u, uRv, v{:}\Box A \Longrightarrow v{:}A}\,\mathsf{g}_{1110}\ w \text{ fresh}}{x{\leq}y, yRz, z{\leq}v, y{\leq}u, uRv, v{:}\Box A \Longrightarrow v{:}A}\,\mathsf{F}_1\ u \text{ fresh}}{x{\leq}y, yRz, z{\leq}v, v{:}\Box A \Longrightarrow v{:}A}\,\supset^{\circ}\ v \text{ fresh}}{x{\leq}y, yRz \Longrightarrow z{:}\Box A \supset A}\,\Box^{\circ}}{\Longrightarrow x{:}\Box(\Box A \supset A)} \tag{4.8}$$

We can then show that Theorem 4.2.2 generalizes to $\mathsf{labIK}_{\leq}$ with any $\mathsf{g_{klmn}}$ rule (it can be one or more) to provide a sound and cut-free complete system for this family of logics.

Theorem 4.6.1. *Let $\mathcal{A}_{\mathsf{g_{klmn}}}$ a set of one-sided Scott-Lemmon axioms and let $\mathcal{R}_{\mathsf{g_{klmn}}}$ be the set of their corresponding $\mathsf{g_{klmn}}$ rules. For any formula B, the following are equivalent.*

1. *B is a theorem of $\mathsf{IK} + \mathcal{A}_{\mathsf{g_{klmn}}}$.*
2. *B is provable in $\mathsf{labIK}_{\leq} + \mathcal{R}_{\mathsf{g_{klmn}}} + \mathsf{cut}$.*
3. *B is provable in $\mathsf{labIK}_{\leq} + \mathcal{R}_{\mathsf{g_{klmn}}}$.*
4. *B is valid in every bi-relational frame satisfying the klmn-incestuality properties.*

Proof. The proof is similar to the one of Theorem 4.2.2.

- 1 $\Longrightarrow$ 2: Same as Theorem 4.3.1 with the additional derivation of $\Diamond^k\Box^l A \supset \Box^m\Diamond^n A$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{y_k{:}\Box^l A, w{:}A \Longrightarrow x_m{:}\Diamond^n A, w{:}A}\,\mathsf{id}}{y_k{\leq}y''_k, \ldots, w{\leq}w, y_k{:}\Box^l A \Longrightarrow x_m{:}\Diamond^n A, w{:}A}\,\Box^{\bullet}}{y_k{:}\Box^l A \Longrightarrow x_m{:}\Diamond^n A, w{:}A}\,{\leq}\mathsf{tr}{+}{\leq}\mathsf{rf}}{y'_k{\leq}y''_k, y''_k R^l w, x_m R^n w, y_k{:}\Box^l A \Longrightarrow x_m{:}\Diamond^n A}\,\Diamond^{\circ}}{x''_0 R y'_1, \{y'_j R y'_{j+1}\}_{1\leq j\leq k-1}, y_k{:}\Box^l A \Longrightarrow x_m{:}\Diamond^n A}\,\mathsf{g_{klmn}}}{x_0 R y_1, \{y_j R y_{j+1}\}_{1\leq j\leq k-1}, y_k{:}\Box^l A \Longrightarrow x_m{:}\Diamond^n A}\,\mathsf{F}_2}{x'_{m-2}{\leq}x''_{m-2}, x''_{m-2} R x'_{m-1}, \{x'_i{\leq}x''_i, x''_i R x''_{i+1}\}_{0\leq i\leq m-2}, x_0{:}\Diamond^k\Box^l A \Longrightarrow x_m{:}\Diamond^n A}\,\Diamond^{\bullet}}{\{x_i{\leq}x'_i, x'_i R x_{i+1}\}_{0\leq i\leq m-1}, x_0{:}\Diamond^k\Box^l A \Longrightarrow x_m{:}\Diamond^n A}\,\mathsf{F}_1}{x{\leq}x_0, x_0{:}\Diamond^k\Box^l A \Longrightarrow x_0{:}\Box^m\Diamond^n A}\,\Box^{\circ}}{\Longrightarrow x{:}\Diamond^k\Box^l A \supset \Box^m\Diamond^n A}\,\supset^{\circ}$$

  where we omit the accumulated relational context for space reason.

- 2 $\Longrightarrow$ 3: To prove that the rule cut is admissible for $\mathsf{labIK}_{\leq} + \mathsf{g_{klmn}}$, it is enough to insert a case for the rule $\mathsf{g_{klmn}}$ in the proof of Theorem 4.2.2, which is straightforward as the $\mathsf{g_{klmn}}$ rule only manipulates the relational context as it shows below:

$$\dfrac{\dfrac{\dfrac{\mathcal{D}_1\,\|}{\mathcal{R}, xR^ky, xR^mz, y{\leq}y', y'R^lu, zR^nu, \Gamma \Longrightarrow \Delta, z{:}C}\,\mathsf{g_{klmn}}}{\mathcal{R}, xR^ky, xR^mz, \Gamma \Longrightarrow \Delta, z{:}C}\,y', u \text{ fresh} \quad \dfrac{\mathcal{D}_2\,\|}{\mathcal{R}, xR^ky, xR^mz, \Gamma, z{:}C \Longrightarrow \Delta}}{\mathcal{R}, xR^ky, xR^mz, \Gamma \Longrightarrow \Delta}\,\mathsf{cut}$$

$$\rightsquigarrow \quad \dfrac{\dfrac{\mathcal{D}_1\,\|}{\mathcal{R}, xR^ky, xR^mz, y{\leq}y', y'R^lu, zR^nu, \Gamma \Longrightarrow \Delta, z{:}C} \quad \dfrac{\mathcal{D}_2^{\mathsf{w}}\,\|}{\mathcal{R}, xR^ky, xR^mz, y{\leq}y', y'R^lu, zR^nu, \Gamma, z{:}C \Longrightarrow \Delta}}{\dfrac{\mathcal{R}, xR^ky, xR^mz, y{\leq}y', y'R^lu, zR^nu, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xR^ky, xR^mz, \Gamma \Longrightarrow \Delta}\,\mathsf{g_{klmn}}\,y', u \text{ fresh}}\,\mathsf{cut}$$

- 3 $\Longrightarrow$ 4: As we already proved the rules of $\mathsf{labIK}_{\leq}$ sound in Theorem 4.4.3, we only need to prove that $\mathsf{g_{klmn}}$ is sound. By way of contradiction, assume that $\mathcal{R}, y{\leq}y', xR^ky, xR^mz, y'R^lu, zR^nu, \Gamma \Longrightarrow \Delta$ is valid in any klmn-incestuous frame, but that there is such a model $\mathcal{M}$ and an interpretation $[\![\cdot]\!]$, such that $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, xR^ky, xR^mz, \Gamma \Longrightarrow \Delta$. That means, $[\![x]\!]R^k_{\mathcal{M}}[\![y]\!]$, $[\![x]\!]R^m_{\mathcal{M}}[\![z]\!]$, for all $x{:}A \in \Gamma$, $\mathcal{M}, x \Vdash A$, and for all $w{:}B \in \Delta$, $\mathcal{M}, w \nVdash B$. Since $\mathcal{M}$ is klmn-incestuous, there exists $v, w \in W_{\mathcal{M}}$, such that $[\![y]\!] \leq_{\mathcal{M}} v$, $vR^l_{\mathcal{M}}w$, and $[\![z]\!]R^n_{\mathcal{M}}w$. Now let $[\![\cdot]\!]^*$ be the extension of $[\![\cdot]\!]$ such that $[\![y']\!]^* = v$, $[\![u]\!]^* = w$, and $[\![\cdot]\!]^* = [\![\cdot]\!]$ otherwise. Then, $\mathcal{M}, [\![\cdot]\!]^* \nVdash \mathcal{R}, y{\leq}y', xR^ky, xR^mz, y'R^lu, zR^nu, \Gamma \Longrightarrow \Delta$. Contradiction.

The proof is completed by appealing to Theorem 3.2.2 used as 4 $\Longrightarrow$ 1 to close the equivalence. □

As we realize this theorem might look rather abstract, we reconsider an example that was problematic in previous approaches to the logic $\mathsf{IK} + \Diamond\Box A \supset \Box\Diamond A$, corresponding to the 1111-condition (see details of these approaches in Section 6.3 of Simpson's PhD thesis [59]). Indeed, the formula

$$\Diamond(\Box(a \vee b) \wedge \Diamond a) \wedge \Diamond(\Box(a \vee b) \wedge \Diamond b)) \supset \Diamond(\Diamond a \wedge \Diamond b)$$

is not a theorem of this logic, but would become provable if we directly add to our system the rule

$$\dfrac{\mathcal{R}, wRv, vRu, vRx, uRx, \Gamma \Longrightarrow \Delta}{\mathcal{R}, wRv, vRu, \Gamma \Longrightarrow \Delta}$$

corresponding to the directedness condition $\forall xyz.((xRy \wedge xRz) \supset \exists u.(yRu \wedge zRu))$. By representing bi-relational semantics precisely with both R and $\leq$ relations and by adding the rule $\mathsf{g_{1111}}$ defined above, we can no longer derive this formula, as illustrated by the representation of the failed proof search below:

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdots}{\mathcal{R}_1, \mathcal{R}_2, y':\Box(a\vee b), u'':b, y':\Diamond a, z':\Box(a\vee b), v'':a, z':\Diamond b \Longrightarrow x':\Diamond(\Diamond a\wedge\Diamond b), y':\Diamond b, v'':b, z':\Diamond a, u'':a}}{\mathcal{R}_1, \mathcal{R}_2, y':\Box(a\vee b), u'':b, y':\Diamond a, z':\Box(a\vee b), v'':a\vee b, z':\Diamond b \Longrightarrow x':\Diamond(\Diamond a\wedge\Diamond b), y':\Diamond b, v'':b, z':\Diamond a, u'':a}\ \vee^\bullet,\mathsf{id}}{\mathcal{R}_1, \mathcal{R}_2, y':\Box(a\vee b), u'':b, y':\Diamond a, z':\Box(a\vee b), z':\Diamond b \Longrightarrow x':\Diamond(\Diamond a\wedge\Diamond b), y':\Diamond b, z':\Diamond a, u'':a}\ \Box^\bullet,\Diamond^\circ}{\mathcal{R}_1, y':\Box(a\vee b), u'':b, y':\Diamond a, z':\Box(a\vee b), z':\Diamond b \Longrightarrow x':\Diamond(\Diamond a\wedge\Diamond b), y':\Diamond b, z':\Diamond a, u'':a}\ \mathsf{g}_{1111}\ z'', v''\text{ fresh}}{\mathcal{R}_1, y':\Box(a\vee b), u'':a\vee b, y':\Diamond a, z':\Box(a\vee b), z':\Diamond b \Longrightarrow x':\Diamond(\Diamond a\wedge\Diamond b), y':\Diamond b, z':\Diamond a, u'':a}\ \vee^\bullet,\mathsf{id}}{\mathcal{R}_1, y':\Box(a\vee b), y':\Diamond a, z':\Box(a\vee b), z':\Diamond b \Longrightarrow x':\Diamond(\Diamond a\wedge\Diamond b), y':\Diamond b, z':\Diamond a}\ \Box^\bullet,\Diamond^\circ}{x{\leq}x', x'Ry', x'Rz', y':\Box(a\vee b), y':\Diamond a, z':\Box(a\vee b), z':\Diamond b \Longrightarrow x':\Diamond(\Diamond a\wedge\Diamond b), y':\Diamond b, z':\Diamond a}\ \mathsf{g}_{1111}\ y'', u''\text{ fresh}}{x{\leq}x', x'Ry', x'Rz', y':\Box(a\vee b), y':\Diamond a, z':\Box(a\vee b), z':\Diamond b \Longrightarrow x':\Diamond(\Diamond a\wedge\Diamond b), y':\Diamond b, z':\Diamond a\wedge\Diamond b}\ \wedge^\circ,\mathsf{id}}{x{\leq}x', x'Ry', y':\Box(a\vee b), y':\Diamond a, x':\Diamond(\Box(a\vee b)\wedge\Diamond b) \Longrightarrow x':\Diamond(\Diamond a\wedge\Diamond b), y':\Diamond b}\ \Diamond^\bullet,\wedge^\bullet,\Diamond^\circ\ z'\text{ fresh}}{x{\leq}x', x'Ry', y':\Box(a\vee b), y':\Diamond a, x':\Diamond(\Box(a\vee b)\wedge\Diamond b) \Longrightarrow x':\Diamond(\Diamond a\wedge\Diamond b), y':\Diamond a\wedge\Diamond b}\ \wedge^\circ,\mathsf{id}}{x{\leq}x', x':\Diamond(\Box(a\vee b)\wedge\Diamond a), x':\Diamond(\Box(a\vee b)\wedge\Diamond b) \Longrightarrow x':\Diamond(\Diamond a\wedge\Diamond b)}\ \Diamond^\bullet,\wedge^\bullet,\Diamond^\circ\ y'\text{ fresh}}{\Longrightarrow x:(\Diamond(\Box(a\vee b)\wedge\Diamond a)\wedge\Diamond(\Box(a\vee b)\wedge\Diamond b))\supset\Diamond(\Diamond a\wedge\Diamond b)}\ \supset^\circ,\wedge^\bullet
$$

where $\mathcal{R}_1 = x{\leq}x', y'{\leq}y'', x'Ry', x'Rz', y''Ru'', z'Ru''$ and $\mathcal{R}_2 = z'{\leq}z'', z''Rv'', y'Rv''$.

As another illustration of our system, let us go back to the example of the transitivity frame condition as it was done in the previous section. The frames that validate the axiom $\mathsf{4}^\Diamond : \Diamond\Diamond A \supset \Diamond A$ are exactly the ones satisfying the 2001-condition: if wRv and vRu, there exists u' such that $u{\leq}u'$ and wRu'. The frames that validate the axiom $\mathsf{4}^\Box : \Box A \supset \Box\Box A$ are exactly the ones satisfying the 0120-condition: if wRv and vRu, there exists w' such that $w{\leq}w'$ and $w'Ru$. We can therefore obtain a sound and complete proof system for the intuitionistic modal logic IK extended with axiom $\mathsf{4}^\Diamond$ or $\mathsf{4}^\Box$, respectively, by specialising the rule scheme above as

$$
\mathsf{g}_{2001}\ \dfrac{\mathcal{R}, wRv, vRu, u{\leq}u', wRu', \Gamma \Longrightarrow \Delta}{\mathcal{R}, wRv, vRu, \Gamma \Longrightarrow \Delta}\ u'\text{ fresh} \qquad \text{or} \qquad \mathsf{g}_{0120}\ \dfrac{\mathcal{R}, wRv, vRu, w{\leq}w', w'Ru, \Gamma \Longrightarrow \Delta}{\mathcal{R}, wRv, vRu, \Gamma \Longrightarrow \Delta}\ w'\text{ fresh}
$$

Indeed, they allow us to derive the corresponding axioms $\mathsf{4}^\Diamond$ and $\mathsf{4}^\Box$ as required above in the completeness proof

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x{\leq}w, wRv, vRu, u{\leq}u', wRu', u:A \Longrightarrow w:\Diamond A, u':A}\ \mathsf{id}}{x{\leq}w, wRv, vRu, u{\leq}u', wRu', u:A \Longrightarrow w:\Diamond A}\ \Diamond^\circ}{x{\leq}w, wRv, vRu, u:A \Longrightarrow w:\Diamond A}\ \mathsf{g}_{2001}}{x{\leq}w, w:\Diamond\Diamond A \Longrightarrow w:\Diamond A}\ \Diamond^\bullet\ (\text{double line})}{x:\Diamond\Diamond A \supset \Diamond A}\ \supset^\circ
$$

as well as,

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x{\leq}w, w{\leq}w', w'Rv, v{\leq}v', v'Ru, w'{\leq}t, tRv', w{\leq}t, tRu, w:\Box A, u:\mathcal{A} \Longrightarrow u:A}\ \mathsf{id}}{x{\leq}w, w{\leq}w', w'Rv, v{\leq}v', v'Ru, w'{\leq}t, tRv', t{\leq}t', t'Ru, w{\leq}t, w:\Box A \Longrightarrow u:A}\ \Box^\bullet}{x{\leq}w, w{\leq}w', w'Rv, v{\leq}v', v'Ru, w'{\leq}t, tRv', t{\leq}t', t'Ru, w:\Box A \Longrightarrow u:A}\ {\leq}\mathsf{tr}}{x{\leq}w, w{\leq}w', w'Rv, vRv', v'Ru, w'{\leq}t, tRv', w:\Box A \Longrightarrow u:A}\ \mathsf{g}_{0120}}{x{\leq}w, w{\leq}w', w'Rv, v{\leq}v', v'Ru, w:\Box A \Longrightarrow u:A}\ \mathsf{F}_1}{x{\leq}w, w:\Box A \Longrightarrow w:\Box\Box A}\ \Box^\circ\ (\text{double line})}{x:\Box A \supset \Box\Box A}\ \supset^\circ
$$

4.6.2 Path Scott-Lemmon axioms

We then also consider the logics defined by a particular subclass of the Scott-Lemmon axioms in which we can find simpler and more elegant extensions for our fully labelled sequent system $\mathsf{labIK}_{\leq}$. In particular, we show that we can extend $\mathsf{labIK}_{\leq}$ with only one rule for each corresponding axiom and avoiding the preorder relation in such rules. This subclass of axioms includes the axiom $\mathsf{d} : \Box A \supset \Diamond A$ and the *path Scott-Lemmon axioms* presented in Chapter 3, Section 3.2 which are of the form:

$$(\Diamond^k \Box A \supset \Box^m A) \wedge (\Diamond^m A \supset \Box^k \Diamond A)$$

where k, m are natural numbers. The path Scott-Lemmon axioms satisfy the *intuitionistic* SL*-path condition* presented in 3.2.

Let $\mathcal{A}$ be the set of axioms including the path Scott-Lemmon axioms and the axiom d, i.e. $\mathcal{A} \subseteq \{\mathsf{t}, \mathsf{b}, \mathsf{4}, \mathsf{5}, \mathsf{d}\}$. In order to capture the axiom of seriality d, we extend the system $\mathsf{labIK}_{\leq}$ by adding the following rule called $R\mathsf{ser}$:

$$R\mathsf{ser}\ \frac{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta}{\mathcal{R}, \Gamma \Longrightarrow \Delta}\ y \text{ fresh}$$

where x is a label presented in $\mathcal{R}$.

In the case of path Scott-Lemmon axioms, we follow again the axiom-as-rule idea, to have a sound a complete system for IK extended by any such axiom and we introduce to the system $\mathsf{labIK}_{\leq}$ the following rule called ϕ_{km} for any natural numbers k, m:

$$\phi_{km}\ \frac{\mathcal{R}, xR^k y, xR^m z, yRz, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xR^k y, xR^m z, \Gamma \Longrightarrow \Delta}$$

Observe, that in the rule ϕ_{km} above, $xR^k y$ and $xR^m z$ are similar to the ones used in the one-sided Scott-Lemmon axioms. For example, in the case we want to restrict the class of frames with the transitivity condition introduced in the beginning of the Chapter, we will have that $k = 0$ and $m = 2$ and we will reach to the rule called $R\mathsf{tr}$ which will be added to the fully labelled system $\mathsf{labIK}_{\leq}$:

$$R\mathsf{tr}\ \frac{\mathcal{R}, xRy, yRz, xRz, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xRy, yRz, \Gamma \Longrightarrow \Delta}$$

We can then show that the axioms for transitivity $\mathsf{4}^{\Diamond}$: $\Diamond\Diamond A \supset \Diamond A$ and $\mathsf{4}^{\Box}$: $\Box A \supset \Box\Box A$ can be derived using the rules of $\mathsf{labIK}_{\leq} + R\mathsf{tr}$ as follows:

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\mathsf{id_g}}{x{\leq}y, yRz, zRu, yRu, u{:}A \Rightarrow y{:}\Diamond A, u{:}A}\,\Diamond^{\circ}}{x{\leq}y, yRz, zRu, yRu, u{:}A \Rightarrow y{:}\Diamond A}\,R\mathsf{tr}}{x{\leq}y, yRz, zRu, u{:}A \Rightarrow y{:}\Diamond A}\,\Diamond^{\bullet}\ u \text{ fresh}}{x{\leq}y, yRz, z{:}\Diamond A \Rightarrow y{:}\Diamond A}\,\Diamond^{\bullet}\ z \text{ fresh}}{x{\leq}y, y{:}\Diamond\Diamond A \Rightarrow y{:}\Diamond A}\,\supset^{\circ}\ y \text{ fresh}}{\Rightarrow x{:}\Diamond\Diamond A \supset \Diamond A}$$

$$R\mathsf{rf}\ \frac{\mathcal{R}, xRx, \Gamma \Longrightarrow \Delta}{\mathcal{R}, \Gamma \Longrightarrow \Delta} \qquad R\mathsf{tr}\ \frac{\mathcal{R}, xRy, yRz, xRz, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xRy, yRz, \Gamma \Longrightarrow \Delta}$$

$$R\mathsf{euc}\ \frac{\mathcal{R}, xRy, xRz, yRz, \Gamma \Longrightarrow \Delta}{\mathcal{R}, \Gamma, xRy, xRz, \Longrightarrow \Delta} \qquad R\mathsf{sim}\ \frac{\mathcal{R}, yRx, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta}$$

$$R\mathsf{ser}\ \frac{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta}{\mathcal{R}, \Gamma \Longrightarrow \Delta}\ y \text{ fresh}$$

Figure 4.2: R-rules

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{v{\leq}v, x{\leq}y, y{\leq}z, u{\leq}w, z{\leq}u', y{\leq}u', u'Rw, zRu, wRv, u'Rv, y{:}\Box A, v{:}A \Rightarrow v{:}A}\,\mathsf{id_g}}{x{\leq}y, y{\leq}z, u{\leq}w, z{\leq}u', y{\leq}u', u'Rw, zRu, wRv, u'Rv, y{:}\Box A \Rightarrow v{:}A}\,\Box^{\bullet}{+}{\leq}\mathsf{rf}}{x{\leq}y, y{\leq}z, u{\leq}w, z{\leq}u', u'Rw, zRu, wRv, y{:}\Box A \Rightarrow v{:}A}\,{\leq}\mathsf{tr}{+}R\mathsf{tr}}{x{\leq}y, y{\leq}z, u{\leq}w, zRu, wRv, y{:}\Box A \Rightarrow v{:}A}\,\mathsf{F}_1\ u' \text{ fresh}}{x{\leq}y, y{\leq}z, zRu, y{:}\Box A \Rightarrow u{:}\Box A}\,\Box^{\circ}\ w, v \text{ fresh}}{x{\leq}y, y{:}\Box A \Rightarrow y{:}\Box\Box A}\,\Box^{\circ}\ z, u \text{ fresh}}{\Rightarrow x{:}\Box A \supset \Box\Box A}\,\supset^{\circ}\ y \text{ fresh}$$

As we have just seen, we are now able to capture stronger logics by only adding *one* rule for each corresponding axiom in $\mathcal{A} \subseteq \{\mathsf{t}, \mathsf{b}, \mathsf{4}, \mathsf{5}, \mathsf{d}\}$ to our fully labelled system. Figure 4.2 presents the four rules for each path Scott-Lemmon axiom (obtained from ϕ_{km}) and the rule $R\mathsf{ser}$ corresponding to the axiom d. The following table shows each axiom, the frame condition each satisfies and its corresponding rule from Figure 4.2:

Axiom	Frame condition	Corresponding R-rule
$\mathsf{t} : A \supset \Diamond A$	$\forall x.xRx$	$R\mathsf{rf}$
$\mathsf{b} : A \supset \Box\Diamond A$	$\forall x, y.xRy \supset yRx$	$R\mathsf{sim}$
$\mathsf{d} : \Box A \supset \Diamond A$	$\forall x \exists y.xRy$	$R\mathsf{ser}$
$\mathsf{4} : \Diamond\Diamond A \supset \Diamond A$	$\forall x, y, z.(xRy \wedge yRz) \supset xRz$	$R\mathsf{tr}$
$\mathsf{5} : \Diamond A \supset \Box\Diamond A$	$\forall x, y, z.(xRy \wedge xRz) \supset yRz$	$R\mathsf{euc}$

Let $\mathcal{R}_{\phi_{km}}$ be the set of rules containing the rules from Figure 4.2. We can now show that Theorem 4.2.2 generalizes to $\mathsf{labIK}_{\leq}$ with any rule(s) of $\mathcal{R}_{\phi_{km}}$ to provide a sound and cut-free complete system for this family of logics capturing the *path Scott-Lemmon axioms* and the axiom d.

Theorem 4.6.2. *Let $\mathcal{A}$ be the set of axioms $\mathcal{A} \subseteq \{\mathsf{t}, \mathsf{b}, \mathsf{4}, \mathsf{5}, \mathsf{d}\}$ and let $\mathcal{R}_{\phi_{km}}$ be the set of their corresponding rules. For any formula B, the following are equivalent:*

1. *B is a theorem of $\mathsf{IK} + \mathcal{A}$.*

2. *B is provable in* $\mathsf{labIK}_{\leq} + \mathcal{R}_{\phi_{km}} + \mathsf{cut}$.

3. *B is provable in* $\mathsf{labIK}_{\leq} + \mathcal{R}_{\phi_{km}}$.

4. *B is valid in every birelational frame satisfying the corresponding conditions for $\mathcal{A}$.*

Proof. The proof is similar to the one of Theorem 4.2.2.

- $1 \implies 2$: Same as Theorem. 4.3.1 with the additional derivation of the path Scott-Lemmon axiom $(\Diamond^k \Box A \supset \Box^m A) \wedge (\Diamond^m A \supset \Box^k \Diamond A)$ which is capturing the axioms $\{\mathsf{t}, \mathsf{b}, \mathsf{4}, \mathsf{5}\}$. We can derive each of these conjuncts as follows (where we omit the accumulated relational context for space reason):

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathsf{id_g}\ \overline{\phantom{y_k \leq y'_k, y'_k R x_m, y_k{:}\Box A, x_m{:}A \Rightarrow x_m{:}A}}}{\Box^\bullet\ y_k \leq y'_k, y'_k R x_m, y_k{:}\Box A, x_m{:}A \Rightarrow x_m{:}A}}{\leq\mathsf{tr}+\phi_{km}\ \{y_j \leq y'_j\}_{1\leq j\leq k-1}, x''_0 R y'_1, \{y'_j R y'_{j+1}\}_{1\leq j\leq k-1}, y_k{:}\Box A \Rightarrow x_m{:}A}}{\mathsf{F_2}\ x_0 R y_1, \{y_j R y_{j+1}\}_{1\leq j\leq k-1}, y_k{:}\Box A \Longrightarrow x_m{:}A}}{\Diamond^\bullet\ x'_{m-2} \leq x''_{m-2}, x''_{m-2} R x'_{m-1}, \{x'_i \leq x''_i, x''_i R x''_{i+1}\}_{0\leq i\leq m-2}, x_0{:}\Diamond^k \Box A \Longrightarrow x_m{:}A}}{\mathsf{F_1}\ x \leq x_0, \{x_i \leq x'_i, x'_i R x_{i+1}\}_{0\leq i\leq m-1}, x_0{:}\Diamond^k \Box A \Longrightarrow x_m{:}A}}{\Box^\circ\ x \leq x_0, x_0{:}\Diamond^k \Box A \Rightarrow x_0{:}\Box^m A}}{\supset^\circ\ \Rightarrow x{:}\Diamond^k \Box A \supset \Box^m A}$$

Observe that the rule ϕ_{km} is applied to $x''_0 R^k y'_k$ (which is the abbreviation of $x''_0 R y'_1$, $y'_2 R y'_3, ..., y'_{k-1} R y'_k$) and $x''_0 R^m x_m$ (which is the abbreviation of $x''_0 R x''_1, ..., x''_{m-1} R x_m$).

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathsf{id_g}\ \overline{\phantom{y_m \leq y'_m, x_k R y'_m, y_m{:}A \Rightarrow x_k{:}\Diamond A, y'_m{:}A}}}{\Diamond^\circ\ y_m \leq y'_m, x_k R y'_m, y_m{:}A \Rightarrow x_k{:}\Diamond A}}{\leq\mathsf{tr}+\phi_{km}\ \{y_j \leq y'_j\}_{1\leq j\leq m-1}, x''_0 R y'_1, \{y'_j R y'_{j+1}\}_{1\leq j\leq m-1}, y_m{:}A \Rightarrow x_k{:}\Diamond A}}{\mathsf{F_2}\ x_0 R y_1, \{y_j R y_{j+1}\}_{1\leq j\leq m-1}, y_m{:}A \Longrightarrow x_k{:}\Diamond A}}{\Diamond^\bullet\ x'_{k-2} \leq x''_{k-2}, x''_{k-2} R x'_{k-1}, \{x'_i \leq x''_i, x''_i R x''_{i+1}\}_{0\leq i\leq k-2}, x_0{:}\Diamond^m A \Longrightarrow x_k{:}\Diamond A}}{\mathsf{F_1}\ \{x_i \leq x'_i, x'_i R x_{i+1}\}_{0\leq i\leq k-1}, x_0{:}\Diamond^m A \Rightarrow x_k{:}\Diamond A}}{\Box^\circ\ x \leq x_0, x_0{:}\Diamond^m A \Rightarrow x_0{:}\Box^k \Diamond A}}{\supset^\circ\ \Rightarrow x{:}\Diamond^m A \supset \Box^k \Diamond A}$$

Observe that in this case the rule ϕ_{km} is applied to $x''_0 R^m y'_m$ (which is the abbreviation of $x''_0 R y'_1, y'_2 R y'_3, ..., y'_{m-1} R y'_m$) and $x''_0 R^k x_k$ (which is the abbreviation of $x''_0 R x''_1, ..., x''_0 R x_k$).

And we also add the derivation of the axiom $\mathsf{d} : \Box A \supset \Diamond A$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x{\leq}y, z{\leq}z, xRy, y{:}\Box A, z{:}A \Longrightarrow z{:}A}\,\mathsf{id_g}}{x{\leq}y, xRy, y{:}\Box A, z{:}A \Longrightarrow z{:}A}\,{\leq}\mathsf{rf}}{x{\leq}y, xRy, y{:}\Box A, z{:}A \Longrightarrow y{:}\Diamond A}\,\Diamond^{\circ}}{x{\leq}y, xRy, y{:}\Box A \Longrightarrow y{:}\Diamond A}\,\Box^{\bullet}}{x{\leq}y, y{:}\Box A \Longrightarrow y{:}\Diamond A}\,R\mathsf{ser}\ z \text{ fresh}}{\Longrightarrow x{:}\Box A \supset \Diamond A}\,\supset^{\circ}\ y \text{ fresh}$$

- $2 \implies 3$: To prove that the cut rule is admissible for $\mathsf{labIK}_{\leq} + \mathcal{R}_{\phi_{km}}$, it is enough to insert a case for the rule ϕ_{km} (which is capturing the axioms $\mathsf{t}, \mathsf{b}, \mathsf{4}, \mathsf{5}$) and a case for $R\mathsf{ser}$ in the proof of Theorem 4.2.2, which is straightforward as these rules only manipulate the relational context as it shows below:

$$\dfrac{\dfrac{\dfrac{\mathcal{D}_1\,\|\!\|}{\mathcal{R}, xR^k y, xR^m z, yRz, \Gamma \Longrightarrow \Delta, z{:}C}\,\phi_{km}}{\mathcal{R}, xR^k y, xR^m z, \Gamma \Longrightarrow \Delta, z{:}C}\,\mathsf{cut} \qquad \dfrac{\mathcal{D}_2\,\|\!\|}{\mathcal{R}, xR^k y, xR^m z, \Gamma, z{:}C \Longrightarrow \Delta}}{\mathcal{R}, xR^k y, xR^m z, \Gamma \Longrightarrow \Delta}$$

$$\rightsquigarrow \quad \dfrac{\dfrac{\mathcal{D}_1\,\|\!\|}{\mathcal{R}, xR^k y, xR^m z, yRz, \Gamma \Longrightarrow \Delta, z{:}C} \quad \dfrac{\mathcal{D}_2^{\mathsf{w}}\,\|\!\|}{\mathcal{R}, xR^k y, xR^m z, yRz, \Gamma, z{:}C \Longrightarrow \Delta}}{\dfrac{\mathcal{R}, xR^k y, xR^m z, yRz, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xR^k y, xR^m z, \Gamma \Longrightarrow \Delta}\,\phi_{km}}\,\mathsf{cut}$$

for the case of $R\mathsf{ser}$:

$$\dfrac{\dfrac{\dfrac{\mathcal{D}_1\,\|\!\|}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta z{:}C}}{\mathcal{R}, \Gamma \Longrightarrow \Delta, z{:}C}\,R\mathsf{ser}\ y \text{ fresh} \qquad \dfrac{\mathcal{D}_2\,\|\!\|}{\mathcal{R}, \Gamma, z{:}C \Longrightarrow \Delta}}{\mathcal{R}, \Gamma \Longrightarrow \Delta}\,\mathsf{cut}$$

$$\rightsquigarrow \quad \dfrac{\dfrac{\mathcal{D}_1\,\|\!\|}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, z{:}C} \quad \dfrac{\mathcal{D}_2^{w}\,\|\!\|}{\mathcal{R}, xRy, \Gamma, z{:}C \Longrightarrow \Delta}}{\dfrac{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta}{\mathcal{R}, \Gamma \Longrightarrow \Delta}\,R\mathsf{ser}\ y \text{ fresh}}\,\mathsf{cut}$$

- $3 \implies 4$: As we already proved the rules of $\mathsf{labIK}_{\leq}$ are sound in Theorem 4.2.2, we only need to prove that ϕ_{km} is sound. By way of contradiction assume that $\mathcal{R}, xR^k y, xR^m z, yRz, \Gamma \implies \Delta$ is valid in any frame restricted with the intuitionistic SL-path condition, but that there is such a model $\mathcal{M}$ and an interpretation $[\![\cdot]\!]$, such that $\mathcal{M}, [\![\cdot]\!] \not\Vdash \mathcal{R}, xR^k y, xR^m z, \Gamma \implies \Delta$. That means, $[\![x]\!] R^k_{\mathcal{M}} [\![y]\!]$, $[\![x]\!] R^m_{\mathcal{M}} [\![z]\!]$,

for all $x{:}A \in \Gamma$, $\mathcal{M}, x \Vdash A$, and for all $y{:}B \in \Delta$, $\mathcal{M}, y \nVdash B$. Since $\mathcal{M}$ follows path Scott-Lemmon axioms property, then we have that $[\![y]\!] R_{\mathcal{M}} [\![z]\!]$. This means that $\mathcal{M}, [\![\cdot]\!] \nVdash \mathcal{R}, xR^k y, xR^m z, yRz, \Gamma \Longrightarrow \Delta$ and we reach contradiction from the assumption.

The proof is completed by appealing to Theorem 3.2.3 which is used as $4 \implies 1$ to close the equivalence. □

As a result of this theorem we can then obtain stronger logics by extending the system $\mathsf{labIK}_{\leq}$ with one (or more) of the rules presented in Figure 4.2 and which represent the Scott-Lemmon axioms of reflexivity ($R\mathsf{rf}$), transitivity ($R\mathsf{tr}$), symmetry ($R\mathsf{sim}$), euclideaness ($R\mathsf{euc}$) and seriality ($R\mathsf{ser}$).

We can now generalize our fully labelled proof system $\mathsf{labIK}_{\leq}$ to a system called $\mathsf{labX}_{\leq}$ where X represents every intuitionistic version of the logics in the $\mathsf{S5}$-cube (Figure 2.2). Depending on the logic we want to capture, we add the corresponding inference rule(s). For example, if we want to capture the logic $\mathsf{X} = \mathsf{IS4}$ (which is reflexive and transitive), we then obtain the system $\mathsf{labIS4}_{\leq}$ by adding $R\mathsf{tr}$ and $R\mathsf{rf}$ to the system $\mathsf{labIK}_{\leq}$, i.e. $\mathsf{labIS4}_{\leq} = \mathsf{labIK}_{\leq} + R\mathsf{tr} + R\mathsf{rf}$.

From Theorem 4.6.2 we get the following Corollary:

Corollary 4.6.3. *A formula A is a theorem of the logic X where $\mathsf{X} \in \{$every intuitionistic modal logic in the $\mathsf{S5}$-cube$\}$ if and only if for every label x, the sequent $\Longrightarrow x{:}A$ is derivable in $\mathsf{labX}_{\leq}$.*

4.7 Conclusion

In this chapter we embrace the fully labelled approach to intuitionistic modal logic with two relation symbols appearing in sequents [43]: one for the accessibility relation associated with the Kripke semantics for normal modal logics and one for the preorder relation associated with the Kripke semantics for intuitionistic logic. This puts our system in close correspondence with the standard birelational Kripke semantics for intuitionistic modal logics. As a consequence it can be extended with arbitrary intuitionistic Scott-Lemmon axioms. Specifically, we expanded it with one-sided Scott-Lemmon axioms and path Scott-Lemmon axioms. This extension allows us to obtain a comprehensive framework that encompasses all fifteen distinct intuitionistic modal logics within the $\mathsf{S5}$-cube. We have established the soundness and completeness of this framework, along with an internal cut elimination proof, encompassing a wider array of intuitionistic modal logics than any existing labelled system.

Chapter 5

Decision procedure for intuitionistic modal logics

As it was mentioned in the introduction, labelled sequents internalize certain elements from Kripke semantics into the syntax of sequents, which turns out to have several interesting consequences to build decision procedures for intuitionistic propositional logic as well as classical modal logic. Our fully labelled proof system $\mathsf{labX}_{\leq}$, presented in the previous chapter, inherits the advantages of labelled systems for both, intuitionistic propositional logic and for classical modal logics: in particular, all inference rules are invertible (i.e. we never delete information bottom-up in proof search) and there is a direct correspondence between sequents and models which lets us build a countermodel by interpreting (and extending) the labelled sequent at which proof search terminates.

In this chapter we use this fully labelled sequent system to give a decision procedure for some intuitionistic modal logics in the $\mathsf{S5}$-cube (Figure 2.2). In particular, we present a search algorithm to prove decidability of the logic X where $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$ and we ensure termination for them. This chapter also serves as a strong introduction to later delve into logics whose decidability was previously unknown, as it is the case with $\mathsf{IS4}$ (see details in Chapter 6). The idea is to apply the same argument presented in this chapter for other logics by adding new loop-checks specially when transitivity is restricting the class of frames.

We develop this chapter by introducing notions that can be used for all the intuitionistic modal logics X in the $\mathsf{S5}$-cube, however, we make distinctions of these logics every time is needed. In Section 5.1, we describe how to read off a model from a sequent and we introduce useful properties of sequents occurring during a proof search in $\mathsf{labX}_{\leq}$. In Section 5.2 we develop concepts necessary to present formally our search algorithm. In Section 5.3 we demonstrate how to retrieve a countermodel from a failed proof search. In Section 5.4, we show how the algorithm produces a proof in $\mathsf{labX}_{\leq}$. We finish our chapter with Section 5.5 showing that the algorithm always terminates for the logics $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$. In Section 5.6 we present our conclusions. We will continue working on the details to reach termination for the logic $\mathsf{IS4}$ in the next chapter.

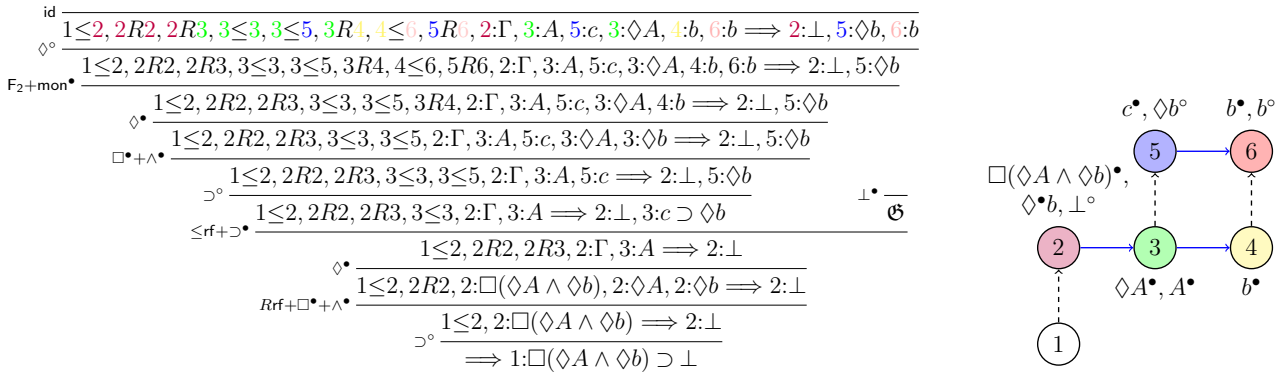


Figure 5.1: On the left: Proof in $\mathsf{labIS4}_{\leq}$ of $\Box(\Diamond A \wedge \Diamond b) \supset \bot$ with $A = (c \supset \Diamond b) \supset \bot$, $\Gamma = \{\Box(\Diamond A \wedge \Diamond b), \Diamond b\}$ and $\mathfrak{G} = 1{\leq}2, 2R2, 2R3, 3{\leq}3, 2{:}\Gamma, 3{:}\bot \Longrightarrow 2{:}\bot$. On the right: Diagrammatic representation of the top left sequent in the derivation.

5.1 Building our search algorithm

Lets us start by introducing the notion of a model corresponding to a sequent, as it allows us to refer to a sequent or of its corresponding model at the same time.

Notation 5.1.1. Let $\mathfrak{G}$ be a sequent $\mathcal{R}, \Gamma \Longrightarrow \Delta$. We use the following notation:

- $x{\leq_{\mathfrak{G}}}y$ iff $x{\leq}y$ occurs in $\mathcal{R}$;
- $xR_{\mathfrak{G}}y$ iff xRy occurs in $\mathcal{R}$;
- $\mathfrak{G}$,$x{:}A^{\bullet}$ iff $x{:}A$ occurs in Γ
  (in this case we also say that $A^{\bullet}$ ***occurs*** at x in $\mathfrak{G}$);
- $\mathfrak{G}$,$x{:}A^{\circ}$ iff $x{:}A$ occurs in Δ
  (in this case we also say that A° ***occurs*** at x in $\mathfrak{G}$).

The use of $\bullet$ and $\circ$ in this way goes back to Lamarche [37]. This notation also allows us to represent sequents graphically, as shown on the right of Figure 5.1 which depicts the top-left sequent of the proof showcased on the left of the Figure (proof of a formula using the fully labelled sequent system $\mathsf{labX}_{\leq}$ where $\mathsf{X} = \mathsf{IS4}$). The sequent is represented by means of a directed graph, whose nodes are the labels of the sequent (for convenience we shall use natural numbers), dashed edges are the $\leq$-relations and solid blue edges are the R-relations. Reflexive and transitive links are omitted in this Figure. We can now simply write a formula $A^{\bullet}$ (respectively A°) next to a label w, to indicated that the labelled formula $w{:}A$ occurs on the left-hand side (respectively right-hand side) of the sequent arrow $\Longrightarrow$.

The following definitions are needed to present our search algorithm to prove decidability of the logic X where X is an intuitionistic modal logic in the $\mathsf{S5}$-cube.

Definition 5.1.2 (X-happy labelled formula)**.** A formula $A^{\bullet}$ (respectively A°) of an intuitionistic modal logic X is X***-happy*** at a label x in a sequent $\mathfrak{G}$, short $\mathfrak{G}$,$x{:}A^{\bullet}$ (respectively $\mathfrak{G}$,$x{:}A^{\circ}$), if and only if the following conditions hold:

- for $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ID}, \mathsf{IK5}, \mathsf{ID5}, \mathsf{IDB}, \mathsf{ITB}\}$:

- $\mathfrak{G}$,x:$a^\bullet$ is always X-happy;
- $\mathfrak{G}$,x:a° is X-happy iff we do not have $\mathfrak{G}$,x:$a^\bullet$;
- $\mathfrak{G}$,x:$\bot^\bullet$ is never X-happy;
- $\mathfrak{G}$,x:$\bot^\circ$ is always X-happy;
- $\mathfrak{G}$,x:$A \wedge B^\bullet$ is X-happy iff $\mathfrak{G}$,x:$A^\bullet$ and $\mathfrak{G}$,x:$B^\bullet$;
- $\mathfrak{G}$,x:$A \wedge B^\circ$ is X-happy iff $\mathfrak{G}$,x:A° or $\mathfrak{G}$,x:B°;
- $\mathfrak{G}$,x:$A \vee B^\bullet$ is X-happy iff $\mathfrak{G}$,x:$A^\bullet$ or $\mathfrak{G}$,x:$B^\bullet$;
- $\mathfrak{G}$,x:$A \vee B^\circ$ is X-happy iff $\mathfrak{G}$,x:A° and $\mathfrak{G}$,x:B°;
- $\mathfrak{G}$,x:$A \supset B^\bullet$ is X-happy iff $\mathfrak{G}$,x:A° or $\mathfrak{G}$,x:$B^\bullet$;
- $\mathfrak{G}$,x:$A \supset B^\circ$ is X-happy iff $\mathfrak{G}$,y:$A^\bullet$ and $\mathfrak{G}$,y:B° for some y such that $x \leq_{\mathfrak{G}} y$;
- $\mathfrak{G}$,x:$\Box A^\bullet$ is X-happy iff $\mathfrak{G}$,z:$A^\bullet$ for all z such that $xR_{\mathfrak{G}}z$;
- $\mathfrak{G}$,x:$\Box A^\circ$ is X-happy iff $\mathfrak{G}$,z:A° for some y, z such that $x \leq_{\mathfrak{G}} y$ and $yR_{\mathfrak{G}}z$;
- $\mathfrak{G}$,x:$\Diamond A^\bullet$ is X-happy iff $\mathfrak{G}$,y:$A^\bullet$ for some y such that $xR_{\mathfrak{G}}y$;
- $\mathfrak{G}$,x:$\Diamond A^\circ$ is X-happy iff $\mathfrak{G}$,y:A° for all y such that $xR_{\mathfrak{G}}y$.

- $\mathsf{X} \in \{\mathsf{IS4}, \mathsf{IK4}, \mathsf{ID4}, \mathsf{ID45}, \mathsf{IK45}, \mathsf{IKB5}, \mathsf{IS5}\}$:
  - $\mathfrak{G}$,x:$\Box A^\bullet$ is X-happy iff $\mathfrak{G}$,z:$A^\bullet$ and $\mathfrak{G}$,z:$\Box A^\bullet$ for all z such that $xR_{\mathfrak{G}}z$;
  - $\mathfrak{G}$,x:$\Diamond A^\circ$ is X-happy iff $\mathfrak{G}$,y:A° and $\mathfrak{G}$,y:$\Diamond A^\circ$ for all y such that $xR_{\mathfrak{G}}y$;
  - all the other cases are the same as for the other intuitionistic modal logics.

Otherwise, we say that the formula is X-***unhappy***.

Observe that the distinction in Definition 5.1.2 is done for the logics containing the $\mathsf{4}$ axiom. This axiom preserves the $\Box^\bullet$ and $\Diamond^\circ$ into the children.

Definition 5.1.3 (X-happy label)**.** A label x occurring in a sequent $\mathfrak{G}$ is X-***happy*** if and only if all formulas occurring at x in $\mathfrak{G}$ are X-happy. A label x is X-***almost happy*** if and only if all formulas occurring at x are X-happy except, possibly, those of the shapes $\bot^\bullet$, a°, $A \supset B^\circ$, and $\Box A^\circ$. A label x is X-***naively happy*** if and only if all formulas occurring at x are X-happy except, possibly, those of the shapes $\bot^\bullet$, $\Diamond A^\bullet$, a°, $A \supset B^\circ$, and $\Box A^\circ$.

The following definition is different for every intuitionistic modal logic X in the $\mathsf{S5}$-cube, since it demands that the structure described by the relational atoms of the sequent has the property that it is demanded from the frame of the model. We define the notion of X-***structurally saturated sequent*** for every logic X as follows:

Definition 5.1.4 (X-structurally saturated sequent)**.** Let X be an intuitionistic modal logic in the $\mathsf{S5}$-cube. A sequent $\mathfrak{G}$ is X-***structurally saturated*** iff for all labels x, y, z and a formula C in $\mathfrak{G}$, the following conditions are satisfied:

- for all logic X:

($\mathsf{mon}^\bullet$) if $x \leq_{\mathfrak{G}} y$ and $\mathfrak{G}$,x:$C^\bullet$, then $\mathfrak{G}$,y:$C^\bullet$;

(F_1) if $x R_{\mathfrak{G}} y$ and $y \leq_{\mathfrak{G}} z$, then there is u such that $x \leq_{\mathfrak{G}} u$ and $u R_{\mathfrak{G}} z$;

(F_2) if $x R_{\mathfrak{G}} y$ and $x \leq_{\mathfrak{G}} z$, then there is u such that $y \leq_{\mathfrak{G}} u$ and $z R_{\mathfrak{G}} u$;

($\leq\mathsf{tr}$) if $x \leq_{\mathfrak{G}} y$ and $y \leq_{\mathfrak{G}} z$, then $x \leq_{\mathfrak{G}} z$;

($\leq\mathsf{rf}$) $x \leq_{\mathfrak{G}} x$ for all x occurring in $\mathfrak{G}$.

- for $\mathsf{X} \in \{\mathsf{IS4}, \mathsf{IK4}, \mathsf{ID4}, \mathsf{ID45}, \mathsf{IK45}, \mathsf{IS5}, \mathsf{IKB5}\}$:

($R\mathsf{tr}$) if $x R_{\mathfrak{G}} y$ and $y R_{\mathfrak{G}} z$, then $x R_{\mathfrak{G}} z$.

- for $\mathsf{X} \in \{\mathsf{IT}, \mathsf{IS4}, \mathsf{ITB}, \mathsf{IS5}\}$:

($R\mathsf{rf}$) $x R_{\mathfrak{G}} x$ for all x occurring in $\mathfrak{G}$.

- for $\mathsf{X} \in \{\mathsf{IKB}, \mathsf{ITB}, \mathsf{IDB}, \mathsf{IKB5}, \mathsf{IS5}\}$:

($R\mathsf{sim}$) if $x R_{\mathfrak{G}} y$, then $y R_{\mathfrak{G}} x$.

- for $\mathsf{X} \in \{\mathsf{ID}, \mathsf{ID5}, \mathsf{ID4}, \mathsf{ID45}, \mathsf{IDB}, \mathsf{IT}, \mathsf{ITB}, \mathsf{IS4}, \mathsf{IS5}\}$:

($R\mathsf{ser}$) for all x occurring in $\mathfrak{G}$, there is a y such that $x R_{\mathfrak{G}} y$.

- for $\mathsf{X} \in \{\mathsf{IK5}, \mathsf{ID5}, \mathsf{IS5}, \mathsf{IK45}, \mathsf{ID45}, \mathsf{IKB5}\}$:

($R\mathsf{euc}$) if $x R_{\mathfrak{G}} y$ and $x R_{\mathfrak{G}} z$, then $y R_{\mathfrak{G}} z$.

Observe that X is defined in each case according to the axioms that the logic X contains, i.e. if the logic contains the axiom $\mathsf{4}$, t, b, d or $\mathsf{5}$ respectively. From this definition, for instance, for the logic IK, we say that a sequent $\mathfrak{G}$ is IK-structurally saturated if and only if it satisfies the conditions $\mathsf{mon}^\bullet$, F_1, F_2, $\leq\mathsf{tr}$, $\leq\mathsf{rf}$. While in the case of the intuitionistic modal logic $\mathsf{IS4}$ in which the frame conditions are restricted with the axioms t and $\mathsf{4}$ (i.e. the birelational models are transitive and reflexive), a sequent $\mathfrak{G}$ is $\mathsf{IS4}$-structurally saturated if and only if it satisfies the conditions for IK and the transitive condition $R\mathsf{tr}$ and the reflexivity condition $R\mathsf{rf}$.

Definition 5.1.5 (X-happy sequent)**.** A sequent $\mathfrak{G}$ is X***-happy*** if and only if it is X-structurally saturated and all labels in the sequent are X-happy.

Definition 5.1.6 (Model of a sequent)**.** Let $\mathfrak{G}$ be a sequent. We define the ***model*** $\mathcal{M}_{\mathfrak{G}}$ ***of*** $\mathfrak{G}$ to be the quadruple $\mathcal{M}_{\mathfrak{G}} = \langle W, \leq_{\mathfrak{G}}, R_{\mathfrak{G}}, V \rangle$ where

- W is the set of labels occurring in $\mathfrak{G}$,
- $\leq_{\mathfrak{G}}$ and $R_{\mathfrak{G}}$ are binary relations between labels of $\mathfrak{G}$ (as defined in Notation 5.1.1), and
- V is a valuation function $V \colon W \to 2^{\mathcal{A}}$ such that for all atoms $a \in \mathcal{A}$ we have $a \in V(w)$ iff $\mathfrak{G}$,$w{:}a^\bullet$.

In the following Theorem, we are going to talk about X-birelational models, in which depending on the logic X we are using, the class of frames will be restricted differently.

Theorem 5.1.7 (Completeness)**.** *For an* X*-happy sequent* $\mathfrak{G}$*, its model* $\mathcal{M}_{\mathfrak{G}} = \langle W, \leq_{\mathfrak{G}}, R_{\mathfrak{G}}, V\rangle$ *is a* X*-birelational model with the following two properties:*

- *if* $\mathfrak{G}$,$x{:}A^{\bullet}$*, then* $\mathcal{M}_{\mathfrak{G}}, x \Vdash A$*;*
- *if* $\mathfrak{G}$,$x{:}A^{\circ}$*, then* $\mathcal{M}_{\mathfrak{G}}, x \nVdash A$*.*

Proof. The worlds W of $\mathcal{M}_{\mathfrak{G}}$ are the labels of the sequent. Conditions F_1 and F_2, the transitivity and reflexivity of $\leq_{\mathfrak{G}}$, and the monotonicity of V all follow by construction due to structural saturation. Furthermore, since the sequent $\mathfrak{G}$ is X-structurally saturated, the frame conditions for every logic X are also satisfied. It only remains to show that for every world $x \in W$ and every formula A, if $\mathfrak{G}$,$x{:}A^{\bullet}$ then $\mathcal{M}_{\mathfrak{G}}, x \Vdash A$ and if $\mathfrak{G}$,$x{:}A^{\circ}$ then $\mathcal{M}_{\mathfrak{G}}, x \nVdash A$. We show this claim by induction on the size of A. We proceed by case analysis on the main connective of A:

- $\mathfrak{G}$,$x{:}\bot^{\bullet}$: it is not possible for a X-happy sequent.
- $\mathfrak{G}$,$x{:}\bot^{\circ}$: we have $\mathcal{M}_{\mathfrak{G}}, x \nVdash \bot$ by definition.
- $\mathfrak{G}$,$x{:}a^{\bullet}$: by Definition 5.1.6, $\mathcal{M}_{\mathfrak{G}}, x \Vdash a$.
- $\mathfrak{G}$,$x{:}a^{\circ}$: it is not the case that $\mathfrak{G}$,$x{:}a^{\bullet}$ by happiness of x, hence, $\mathcal{M}_{\mathfrak{G}}, x \nVdash a$ by Definition 5.1.6.
- $\mathfrak{G}$,$x{:}B \wedge C^{\bullet}$: by happiness of x, we have both $\mathfrak{G}$,$x{:}B^{\bullet}$ and $\mathfrak{G}$,$x{:}C^{\bullet}$. Then, by IH we have $\mathcal{M}_{\mathfrak{G}}, x \Vdash B$ and $\mathcal{M}_{\mathfrak{G}}, x \Vdash C$. Therefore, by definition of forcing we have $\mathcal{M}_{\mathfrak{G}}, x \Vdash B \wedge C$.
- $\mathfrak{G}$,$x{:}B \wedge C^{\circ}$: by happiness of x, we then either have $\mathfrak{G}$,$x{:}B^{\circ}$ or $\mathfrak{G}$,$x{:}C^{\circ}$. Then either $\mathcal{M}_{\mathfrak{G}}, x \nVdash B$ or $\mathcal{M}_{\mathfrak{G}}, x \nVdash C$ by IH. Therefore, $\mathcal{M}_{\mathfrak{G}}, x \nVdash B \wedge C$.
- $\mathfrak{G}$,$x{:}B \vee C^{\bullet}$: by happiness of x, we then either have $\mathfrak{G}$,$x{:}B^{\bullet}$ or $\mathfrak{G}$,$x{:}C^{\bullet}$. Then, by IH we either have $\mathcal{M}_{\mathfrak{G}}, x \Vdash B$ or $\mathcal{M}_{\mathfrak{G}}, x \Vdash C$. By definition of forcing, we have $\mathcal{M}_{\mathfrak{G}}, x \Vdash B \vee C$.
- $\mathfrak{G}$,$x{:}B \vee C^{\circ}$: by happiness of x, we have $\mathfrak{G}$,$x{:}B^{\circ}$ and $\mathfrak{G}$,$x{:}C^{\circ}$. By IH, we have $\mathcal{M}_{\mathfrak{G}}, x \nVdash B$ and $\mathcal{M}_{\mathfrak{G}}, x \nVdash C$. Therefore, we have $\mathcal{M}_{\mathfrak{G}}, x \nVdash B \vee C$.
- $\mathfrak{G}$,$x{:}B \supset C^{\bullet}$: consider any y with $x \leq_{\mathfrak{G}} y$. By ($\mathsf{mon}^{\bullet}$)-structural saturation, we have $\mathfrak{G}$,$y{:}B \supset C^{\bullet}$. By happiness of y, either we have $\mathfrak{G}$,$y{:}B^{\circ}$ or $\mathfrak{G}$,$y{:}C^{\bullet}$. By IH, we then either have $\mathcal{M}_{\mathfrak{G}}, y \nVdash B$ or $\mathcal{M}_{\mathfrak{G}}, y \Vdash C$. Thus, $\mathcal{M}_{\mathfrak{G}}, y \Vdash B$ implies $\mathcal{M}_{\mathfrak{G}}, y \Vdash C$ for all y with $x \leq_{\mathfrak{G}} y$. Therefore, $\mathcal{M}_{\mathfrak{G}}, x \Vdash B \supset C$.
- $\mathfrak{G}$,$x{:}B \supset C^{\circ}$: by happiness of x, there is a world y such that $x \leq_{\mathfrak{G}} y$, $\mathfrak{G}$,$y{:}B^{\bullet}$, and $\mathfrak{G}$,$y{:}C^{\circ}$. By IH, we have $\mathcal{M}_{\mathfrak{G}}, y \Vdash B$ and $\mathcal{M}_{\mathfrak{G}}, y \nVdash C$. This means there is no world y such that $x \leq_{\mathfrak{G}} y$ and $\mathcal{M}_{\mathfrak{G}}, y \Vdash B$ implies $\mathcal{M}_{\mathfrak{G}}, y \Vdash C$. Therefore, $\mathcal{M}_{\mathfrak{G}}, x \nVdash B \supset C$.
- $\mathfrak{G}$,$x{:}\Diamond B^{\bullet}$: by happiness of x, there is a world y such that $x R_{\mathfrak{G}} y$ and $\mathfrak{G}$,$y{:}B^{\bullet}$. By IH, we have $\mathcal{M}_{\mathfrak{G}}, y \Vdash B$ and, therefore, we have $\mathcal{M}_{\mathfrak{G}}, x \Vdash \Diamond B$.
- $\mathfrak{G}$,$x{:}\Diamond B^{\circ}$: by happiness of x, we have $\mathfrak{G}$,$y{:}B^{\circ}$ for all worlds y such that $x R_{\mathfrak{G}} y$. Thus, by IH, $\mathcal{M}_{\mathfrak{G}}, y \nVdash B$ whenever $x R_{\mathfrak{G}} y$. Therefore, $\mathcal{M}_{\mathfrak{G}}, x \nVdash \Diamond B$.

- $\mathfrak{G}$,x:$\Box B^\bullet$: consider arbitrary y and z with $x \leq_{\mathfrak{G}} y$ and $y R_{\mathfrak{G}} z$. By ($\mathsf{mon}^\bullet$)-structural saturation, $\mathfrak{G}$,y:$\Box B^\bullet$. By happiness of y, we have $\mathfrak{G}$,z:$B^\bullet$. Thus, by IH, we have $\mathcal{M}_{\mathfrak{G}}, z \Vdash B$ whenever $x \leq_{\mathfrak{G}} y$ and $y R_{\mathfrak{G}} z$. Therefore, $\mathcal{M}_{\mathfrak{G}}, x \Vdash \Box B$.

- $\mathfrak{G}$,x:$\Box B^\circ$: by happiness of x, there exist y and z such that $x \leq_{\mathfrak{G}} y$, $y R_{\mathfrak{G}} z$, and $\mathfrak{G}$,z:B°. By IH, we have $\mathcal{M}_{\mathfrak{G}}, z \nVdash B$. Therefore, $\mathcal{M}_{\mathfrak{G}}, x \nVdash \Box B$. □

Remark 5.1.8. Observe that in the proof of Theorem 5.1.7 we do not need the special conditions of happiness for $\Diamond^\circ$ and $\Box^\bullet$ introduced in Definition 5.1.2 when the axiom $\mathsf{4}$ is part of the logic. But we will use this notion of happiness in Chapter 6.

Definition 5.1.9 (Axiomatic sequent)**.** A sequent $\mathfrak{G}$ is ***axiomatic*** if and only if there is a label x such that either $\mathfrak{G}$,x:$a^\bullet$ and $\mathfrak{G}$,x:a° for some a, or $\mathfrak{G}$,x:$\bot^\bullet$. Otherwise, $\mathfrak{G}$ is called ***non-axiomatic***.

Remark 5.1.10. An axiomatic sequent $\mathfrak{G}$ is never X-happy, because either:

- it is a $\mathfrak{G}$,x:$\bot^\bullet$ and by definition of happiness it cannot be X-happy, or

- it has $\mathfrak{G}$,x:a° and $\mathfrak{G}$,x:$a^\bullet$, and by definition of happiness $\mathfrak{G}$,x:a° is only X-happy if we do not have $\mathfrak{G}$,x:$a^\bullet$.

All derivations of $\Longrightarrow x{:}A$ obtained by a proof search in system $\mathsf{labX}_{\leq}$ have a particular structure: their labels are partitioned into *layers*, with each layer having a tree structure. This plays an important role in our proof search algorithm so we introduce the following definitions:

Definition 5.1.11 (Layer)**.** For a sequent $\mathfrak{G}$, we define the relation $R^{\leftrightarrow}_{\mathfrak{G}}$ to be the transitive and reflexive closure of $R_{\mathfrak{G}} \cup R^{-1}_{\mathfrak{G}}$. Since this is an equivalence relation, we can define a ***layer*** L in $\mathfrak{G}$ to be an equivalence class of $R^{\leftrightarrow}_{\mathfrak{G}}$.

Definition 5.1.12 (Layered sequent)**.** We say that a sequent $\mathfrak{G}$ is ***layered*** if and only if for any labels x, x', y, and y' occurring in $\mathfrak{G}$:

1. if $x R^{\leftrightarrow}_{\mathfrak{G}} y$ for $x \neq y$, then $x \not\leq_{\mathfrak{G}} y$ and $y \not\leq_{\mathfrak{G}} x$; and

2. if $x R^{\leftrightarrow}_{\mathfrak{G}} y$, $x' R^{\leftrightarrow}_{\mathfrak{G}} y'$, and $x \leq_{\mathfrak{G}} x'$ for $x \neq x'$, then $y' \not\leq_{\mathfrak{G}} y$.

For layers L_1 and L_2, we define $L_1 \leq L_2$ whenever there are labels $x \in L_1$ and $y \in L_2$ such that $x \leq_{\mathfrak{G}} y$. We write $L_1 < L_2$ iff $L_1 \leq L_2$ and $L_1 \neq L_2$.

Proposition 5.1.13. *For a layered X-structurally saturated sequent $\mathfrak{G}$, the relation $\leq$ is an order relation on its layers.*

Proof. For reflexivity, consider any layer L. For any label $x \in L$, by ($\leq\mathsf{rf}$)-structural saturation, $x \leq_{\mathfrak{G}} x$. Hence, $L \leq L$ by Definition 5.1.12.

For transitivity, let $L_1 \leq L_2$ and $L_2 \leq L_3$ for layers L_1, L_2, and L_3. By Definition 5.1.12, $x \leq_{\mathfrak{G}} y'$ and $y \leq_{\mathfrak{G}} z$ for some labels $x \in L_1$, $y', y \in L_2$, and $z \in L_3$. Since $y' R^{\leftrightarrow}_{\mathfrak{G}} y$, by ($\mathsf{F}$)-structural saturation (Proposition 5.1.19), there is a label z' such that $y' \leq_{\mathfrak{G}} z'$ and $z' R^{\leftrightarrow}_{\mathfrak{G}} z$. The latter means that $z' \in L_3$. By ($\leq\mathsf{tr}$)-structural saturation, $x \leq_{\mathfrak{G}} z'$. Hence, $L_1 \leq L_3$ by Definition 5.1.12.

For antisymmetry, assume $L_1 \leq L_2$ and $L_1 \neq L_2$. By Definition 5.1.12, there are labels $x \in L_1$ and $x' \in L_2$ such that $x \leq_{\mathfrak{G}} x'$. For arbitrary labels $y \in L_1$ and $y' \in L_2$, we have $x R^{\leftrightarrow}_{\mathfrak{G}} y$ and $x' R^{\leftrightarrow}_{\mathfrak{G}} y'$. Since $L_1 \neq L_2$, we have $x \neq x'$. Hence, $y' \not\leq_{\mathfrak{G}} y$ by 2) of Definition 5.1.12. Since $y' \not\leq_{\mathfrak{G}} y$ for any $y' \in L_2$ and $y \in L_1$, it is not the case that $L_2 \leq L_1$. □

Note that the second condition in Definition 5.1.12 is only needed to establish antisymmetry in Proposition 5.1.13. If $\leq_{\mathfrak{G}}$ on labels is already antisymmetric then the second condition follows from the first.

In all sequents that we discuss here, the order relation $\leq$ on layers defines a tree structure so we introduce the following definitions:

Definition 5.1.14 (Tree-layered sequent)**.** A layered sequent is ***tree-layered*** if and only if (i) there is a layer L_0 such that $L_0 \leq L$ for all layers L, and (ii) for all layers L, L', and L'', whenever $L' \leq L$ and $L'' \leq L$, then either $L' \leq L''$ or $L'' \leq L'$.

Remark 5.1.15. Note that we do not need to ask for the root of a layer in Definition 5.1.14.

Definition 5.1.16 (X-happy layer)**.** A layer L is ***X-happy/X-almost happy/X-naively happy*** iff all labels of L are X-happy/X-almost happy/X-naively happy respectively.

Remark 5.1.17. Observe that the definitions of X-happy labelled formulas for $A \supset B^{\bullet}$ and $\Box A^{\bullet}$ in Definition 5.1.2 are different from the forcing conditions of the corresponding connectives. This mismatch is intentional with the missing conditions on $\leq$ outsourced to ($\mathsf{mon}^{\bullet}$)-structural saturation (Definition 5.1.4) instead. As a result, happiness becomes almost a local property for layers: if a sequent is modified (by adding labelled formulas or relational atoms) outside of a given X-happy layer, the only formulas whose happiness needs to be reinspected are $A \supset B^{\circ}$ and $\Box A^{\circ}$.

We now introduce the definitions of X-*saturated* and X-*semi-saturated* sequent as our search algorithm works on them.

Definition 5.1.18 (X-saturated and X-semi-saturated sequent)**.** A sequent $\mathfrak{G}$ of a logic X is called ***X-saturated*** if and only if it is tree-layered and X-structurally saturated and all its formulas are X-almost happy. A sequent $\mathfrak{G}$ is ***X-semi-saturated*** if and only if it is tree-layered and X-structurally saturated and all its formulas are X-naively happy. A set $\mathfrak{S}$ of sequents is called ***X-saturated/X-semi-saturated*** if only if it is finite and all elements of $\mathfrak{S}$ are X-saturated/X-semi-saturated, respectively.

Proposition 5.1.19. *If a sequent $\mathfrak{G}$ is X-structurally saturated, then:*

(F) *if $xR^{\leftrightarrow}_{\mathfrak{G}}y$ and $x{\leq_{\mathfrak{G}}}z$, then there is u such that $y{\leq_{\mathfrak{G}}}u$ and $zR^{\leftrightarrow}_{\mathfrak{G}}u$.*

Proof. There must be a sequence $x_0, \ldots, x_n$ of labels with $n \geq 0$ such that $x_0 = x$, $x_n = y$, and for each $0 \leq i \leq n-1$, either $x_i R_{\mathfrak{G}} x_{i+1}$ or $x_{i+1} R_{\mathfrak{G}} x_i$. We use induction on n. If $n = 0$, i.e., $x = y$, then $u := z$ suffices. Otherwise, by IH, there is v such that $x_{n-1}{\leq_{\mathfrak{G}}}v$ and $zR^{\leftrightarrow}_{\mathfrak{G}}v$. If $x_n R_{\mathfrak{G}} x_{n-1}$, there is u by (F_1)-structural saturation such that $x_n{\leq_{\mathfrak{G}}}u$ and $uR_{\mathfrak{G}}v$. If $x_{n-1} R_{\mathfrak{G}} x_n$, there is u by (F_2)-structural saturation such that $x_n{\leq_{\mathfrak{G}}}u$ and $vR_{\mathfrak{G}}u$. Either way, $y{\leq_{\mathfrak{G}}}u$ and $zR^{\leftrightarrow}_{\mathfrak{G}}u$. □

5.2 Search Algorithm

The algorithm that we present here is performing proof search in our fully labelled sequent system $\mathsf{labX}_{\leq}$ for every intuitionistic modal logic X in the S5-cube [1]. Roughly speaking, we

[1] This algorithm does terminate for $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$. In Chapter 6, we will present how to reach termination when the axiom 4 is part of the logic.

work on layers until these layers are saturated. This is achieved by *semi-saturating* the layer, which essentially means that we exhaustively apply all inference rules that do not add new labels to the sequent; and we then pick an X-unhappy $x{:}\Diamond A^\bullet$ and we make it X-happy by applying the $\Diamond^\bullet$-rule. Then we semi-saturate again, and we continue until all $x{:}\Diamond A^\bullet$ in that layer are X-happy, and therefore the layer is saturated.

Then the only X-unhappy formulas in a X-saturated sequent are of the shape $x{:}\bot^\bullet$, $x{:}a^\circ$, $x{:}A \supset B^\circ$ or $x{:}\Box A^\circ$. In the first two cases, the sequent is axiomatic, and we can stop working on it. In the case that all X-unhappy formulas are of the shape $x{:}A \supset B^\circ$ or $x{:}\Box A^\circ$, we proceed by making them X-happy. We take the first $x{:}\Box A^\circ$ or $x{:}A \supset B^\circ$ and make it X-happy by creating a new layer to be saturated. We work on all the remaining X-unhappy $x{:}\Box A^\circ$ and $x{:}A \supset B^\circ$ until they all become X-happy. This process is repeated in the other layers, until we see a *repetition of layers*. In order to achieve termination of this process, we have to implement a loop check looking for a repetition of the sequent in the future branch.

In this section, we formally introduce the concepts needed to understand the details of this search algorithm. As previously indicated, this algorithm has been adapted to work with all the intuitionistic modal logics in the $\mathsf{S5}$-cube. However, to ensure termination in certain logics, additional efforts will need to be presented. In Chapter 6, we show how to deal with the logic $\mathsf{IS4}$ whose decidability was unknown.

5.2.1 Semi-saturation

We begin our search algorithm by making all formulas X-happy that are not of the form $x{:}\bot^\bullet$, $x{:}a^\circ$, $x{:}\Diamond A^\bullet$, $x{:}A \supset B^\circ$ or $x{:}\Box A^\circ$. In other words, we apply all the inference rules that do not create new labels. We called this process *semi-saturation* and it is defined formally as follows:

Definition 5.2.1 (Semi-saturation)**.** Let $\mathcal{G}$ and $\mathcal{G}'$ be sets of sequents and let $\mathfrak{G}$ be a sequent of $\mathcal{G}$. Let X be an intuitionistic modal logic in the $\mathsf{S5}$-cube. We define a binary relation $\rightrightarrows_{\mathsf{s}}$ such that $\mathcal{G} \rightrightarrows_{\mathsf{s}} \mathcal{G}'$ iff there is a sequent $\mathfrak{G} \in \mathcal{G}$ such that some labelled formula C of one of the below shapes occurs at x in $\mathfrak{G}$ and is X-unhappy, and $\mathcal{G}'$ is obtained by replacing sequent $\mathfrak{G}$ in set $\mathcal{G}$, either with sequent $\mathfrak{G}'$ according to cases 1)–5) below, or with sequents $\mathfrak{G}'$ and $\mathfrak{G}''$ according to cases 6)–8) below, as follows:

1. for $C = A \wedge B^\bullet$, obtain $\mathfrak{G}'$ by adding to $\mathfrak{G}$ all of $x{:}A^\bullet$ and $x{:}B^\bullet$ that are missing;
2. for $C = A \vee B^\circ$, obtain $\mathfrak{G}'$ by adding to $\mathfrak{G}$ all of $x{:}A^\circ$ and $x{:}B^\circ$ that are missing;
3. for $C = \Box A^\bullet$, obtain $\mathfrak{G}'$ by adding to $\mathfrak{G}$ all $z{:}A^\bullet$ that are missing whenever $xR_{\mathfrak{G}}z$, and adding $z{:}\Box A^\bullet$ in the cases that is needed[2];
4. for $C = \Diamond A^\circ$, obtain $\mathfrak{G}'$ by adding to $\mathfrak{G}$ all $y{:}A^\circ$ that are missing whenever $xR_{\mathfrak{G}}y$, and adding $y{:}\Diamond A^\circ$ in the cases that is needed[2];
5. for $C = A \vee B^\bullet$, obtain $\mathfrak{G}'$ (resp. $\mathfrak{G}''$) by adding to $\mathfrak{G}$ the labelled formula $x{:}A^\bullet$ (resp. $x{:}B^\bullet$);
6. for $C = A \wedge B^\circ$, obtain $\mathfrak{G}'$ (resp. $\mathfrak{G}''$) by adding to $\mathfrak{G}$ the labelled formula $x{:}A^\circ$ (resp. $x{:}B^\circ$);

[2]See Definition 5.1.2

7. for $C = A \supset B^\bullet$, obtain $\mathfrak{G}'$ (resp. $\mathfrak{G}''$) by adding to $\mathfrak{G}$ the labelled formula $x{:}A^\circ$ (resp. $x{:}B^\bullet$).

We write $\rightsquigarrow_{\mathsf{s}}^*$ for the transitive and reflexive closure of $\rightsquigarrow_{\mathsf{s}}$.If $\mathfrak{G} \rightsquigarrow_{\mathsf{s}} \mathfrak{G}'$ and $\mathfrak{G}'$ is in normal form with respect to $\rightsquigarrow_{\mathsf{s}}$, i.e., if all labels occurring in all sequents from $\mathfrak{G}'$ are X-naively happy, then $\mathfrak{G}'$ is called the ***semi-saturation*** of $\mathfrak{G}$.

Lemma 5.2.2. *The following statements hold:*

a) If a set $\mathfrak{G}$ of sequents is X-structurally saturated and tree-layered and $\mathfrak{G} \rightsquigarrow_{\mathsf{s}} \mathfrak{G}'$, then $\mathfrak{G}'$ is X-structurally saturated and tree-layered.

b) On finite sets $\mathfrak{G}$, the rewrite relation $\rightsquigarrow_{\mathsf{s}}$ is terminating.

Proof. To prove a), assume that the set $\mathfrak{G}$ is finite and all sequents $\mathfrak{G} \in \mathfrak{G}$ are X-structurally saturated and tree-layered. The size of $\mathfrak{G}'$ is larger than that of $\mathfrak{G}$ by at most one sequent, hence, $\mathfrak{G}'$ is also finite. Given that $\mathfrak{G}$ is layered, the labels of all new labelled formulas have no non-reflexive futures, making ($\mathsf{mon}^\bullet$) trivial for them. Since rewrites neither introduce new labels nor add relational atoms, all other conditions of structural saturation, as well as being tree-layered remain true. Therefore, all sequents in $\mathfrak{G}'$ are X-structurally saturated and tree-layered.

To prove b), observe that each rewrite $\rightsquigarrow_{\mathsf{s}}$ adds a labelled formula to a sequent $\mathfrak{G} \in \mathfrak{G}$ if the formula is not already in $\mathfrak{G}$. Moreover, the added formulas are subformulas of formulas in the sequent, and no new labels are introduced. Since each $\mathfrak{G}$ contains finitely many formulas and finitely many labels, only finitely many formulas can be added to the sequent. Thus, if there are finitely many sequents in $\mathfrak{G}$, the relation $\rightsquigarrow_{\mathsf{s}}$ is terminating. □

5.2.2 Saturation

From this step, we work on the intuitionistic modal logics $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$ and we show how to deal with X-unhappy $x{:}\Diamond A^\bullet$. This is the first inference rule which is introducing a new fresh label to the sequent, in this case, in the R-branch. The process of applying the $\Diamond^\bullet$-rule is called *saturation*.

Definition 5.2.3 (Saturation)**.** Let $\mathfrak{G}$ and $\mathfrak{G}'$ be sequents. Let $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$. We define a binary relation $\rightsquigarrow_\Diamond$ on sequents as follows: we have $\mathfrak{G} \rightsquigarrow_\Diamond \mathfrak{G}'$ if and only if:

1. there is a label y and a formula A, such that $\mathfrak{G}$,$y{:}\Diamond A^\bullet$ is X-unhappy, and
2. all labels $u \neq y$ such that $uR_{\mathfrak{G}}y$ but not $yR_{\mathfrak{G}}u$ are X-almost happy,
3. then we obtain $\mathfrak{G}'$ from $\mathfrak{G}$ by adding $z{\leq_{\mathfrak{G}}}z$, $yR_{\mathfrak{G}}z$, $z{:}A^\bullet$ for some fresh label z, and then closing the resulting $R_{\mathfrak{G}}$ under X-structural saturation.

For sets $\mathfrak{G}$ and $\mathfrak{G}'$ of sequents, we write $\mathfrak{G} \rightsquigarrow_\Diamond \mathfrak{G}'$ iff $\mathfrak{G}$ is X-semi-saturated, $\mathfrak{G} \rightsquigarrow_\Diamond \mathfrak{G}'$ for some $\mathfrak{G} \in \mathfrak{G}$ and $\mathfrak{G}'$ is a semi-saturation of $\Big(\mathfrak{G} \setminus \{\mathfrak{G}\}\Big) \cup \{\mathfrak{G}'\}$. We write $\rightsquigarrow_\Diamond^*$ for the transitive and reflexive closure of $\rightsquigarrow_\Diamond$. If $\mathfrak{G} \rightsquigarrow_\Diamond^* \mathfrak{G}'$ and $\mathfrak{G}'$ is in normal form with respect to $\rightsquigarrow_\Diamond$, then $\mathfrak{G}'$ is called a ***saturation*** of $\mathfrak{G}$.

The rewrite relation $\rightrightarrows_{\Diamond}$ does terminate for the mentioned logics IK, IT, IKB, ITB and $\mathsf{IK5}$ but it does not terminate for the other intuitionistic modal logics. We are going to show how to ensure termination for the logic $\mathsf{IS4}$ in Chapter 6.

Lemma 5.2.4. *Let $\mathfrak{G}$ be a X-semi-saturated sequent, let $\mathfrak{G} \rightsquigarrow_{\Diamond} \mathfrak{G}'$, and let $\boldsymbol{\mathfrak{G}}^*$ be a semi-saturation of $\{\mathfrak{G}'\}$. Then all X-happy $\Diamond^\bullet$-formulas are preserved by the transition from $\mathfrak{G}$ to $\boldsymbol{\mathfrak{G}}^*$, i.e., for all sequents $\mathfrak{G}^* \in \boldsymbol{\mathfrak{G}}^*$ and for all labels u occurring in $\mathfrak{G}^*$, if $\mathfrak{G}$,u:$\Diamond C^\bullet$ is X-happy, then so is $\mathfrak{G}^*$,u:$\Diamond C^\bullet$.*

Proof. Note that semi-saturation does not remove formulas or labels, nor does it add R-relational atoms. Hence, semi-saturation cannot spoil happiness of any $\Diamond^\bullet$-formula and it is sufficient to show that no X-happy $\Diamond^\bullet$-formula of $\mathfrak{G}$ becomes X-unhappy in $\mathfrak{G}'$. The rewrite relation $\rightsquigarrow_{\Diamond}$ makes at least one $\Diamond^\bullet$-formula X-happy in $\mathfrak{G}'$ and since nothing is removed in the semi-saturation $\boldsymbol{\mathfrak{G}}^*$, then all X-happy $\Diamond^\bullet$-formulas are preserved. □

Lemma 5.2.5. *Let $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$, the following statements hold:*

a) If a set $\boldsymbol{\mathfrak{G}}$ of sequents is X-semi-saturated and $\boldsymbol{\mathfrak{G}} \rightrightarrows_{\Diamond} \boldsymbol{\mathfrak{G}}'$, then $\boldsymbol{\mathfrak{G}}'$ is X-semi-saturated.

b) A set $\boldsymbol{\mathfrak{G}}$ of sequents is X-saturated iff it is X-semi-saturated and in normal form w.r.t. $\rightrightarrows_{\Diamond}$.

Proof. To prove a, assume that set $\boldsymbol{\mathfrak{G}}$ is finite and all sequents in $\boldsymbol{\mathfrak{G}}$ are X-semi-saturated. Set $\boldsymbol{\mathfrak{G}}' = \big(\boldsymbol{\mathfrak{G}} \setminus \{\mathfrak{G}\}\big) \cup \boldsymbol{\mathfrak{G}}_{\mathfrak{G} \rightsquigarrow_{\Diamond} \mathfrak{G}'}$ for some $\mathfrak{G} \rightsquigarrow_{\Diamond} \mathfrak{G}'$ where $\mathfrak{G} \in \boldsymbol{\mathfrak{G}}$ is X-semi-saturated. The binary relation $\rightsquigarrow_{\Diamond}$ makes an X-unhappy $\Diamond^\bullet$-formula of $\mathfrak{G}$ becomes X-happy in $\mathfrak{G}'$. This procedure is not changing the happiness of the previous formulas, they remain X-semi-saturated, but it is possible that in the new label introduced by $\Diamond^\bullet$ there exists an X-unhappy formula. For this reason, after each $\rightsquigarrow_{\Diamond}$, we apply semi-saturation again. Then $\boldsymbol{\mathfrak{G}}'$ is X-semi-saturated.

To prove b, observe that $\boldsymbol{\mathfrak{G}}$ is X-saturated and by definition it is X-semi-saturated and all the formulas of the form $\Diamond A^\bullet$ occurring on it are X-happy. Saturation $\rightrightarrows_{\Diamond}$ can be applied to a X-semi-saturated set if and only if it has a X-unhappy $\Diamond^\bullet$-formula. Hence, $\boldsymbol{\mathfrak{G}}$ is X-saturated if and only if it is X-semi-saturated and in normal form with respect to $\rightrightarrows_{\Diamond}$. □

Lemma 5.2.6. *Let $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$. The rewrite relation $\rightrightarrows_{\Diamond}$ is terminating.*

Proof. Given a layer L ocurring in a sequent $\mathfrak{G}$ and suppose that L contains some X-unhappy $\Diamond^\bullet$-formulas. Observe that each rewrite $\rightrightarrows_{\Diamond}$ makes one $\Diamond^\bullet$-formula X-unhappy to become X-happy (i.e. it creates a new label in the R-branch and it adds to this label a formula which is a subformula of the formula in the sequent) and it applies semi-saturation immediately after. Observe that the application of the $\Diamond^\bullet$ + semi-saturation does not make any X-happy $\Diamond^\bullet$-formulas X-unhappy (happiness is preserved by Lemma 5.2.4).

Observe that each application of $\Diamond^\bullet$ might only introduce X-unhappy $\Diamond^\bullet$-formulas having smaller modal degree than the $\Diamond^\bullet$-formula to which the rule was applied: suppose that you have a formula F at a specific label x in a sequent $\mathfrak{G}$ of modal degree n. Assume this formula F has X-unhappy $\Diamond^\bullet$-formulas. To make them X-happy, we introduce a new label in the R-branch with a formula whose modal depth is strictly smaller, i.e. the modal depth is $n-1$ (since $\Diamond^\bullet$ is not preserved in the application of $\Diamond^\bullet$-rule).

Thus, every application of a $\Diamond^\bullet$-rule to some labelled formula of modal degree n in L is such that the total number of X-unhappy $\Diamond^\bullet$-formulas of degree n decreases, and only X-unhappy formulas of degree $n-1$ are introduced. In a number of steps which is bounded by the number and modal degree of the $\Diamond^\bullet$-formulas, the rewrite relation $\rightrightarrows_{\Diamond}$ terminates. □

5.2.3 Lifting saturation

After having discussed how to saturate sets of sequents for the logics $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$, we are now going to show how to deal with X-unhappy $A \supset B^\circ$ and $\Box A^\circ$. For this, we have to create new layers and we can just apply the naive strategy: copy the R-structure of the layer that contains the X-unhappy labelled formulas of the shape $x{:}A \supset B^\circ$ or $x{:}\Box A^\circ$.

Definition 5.2.7 (Sequent sum)**.** Let $\mathfrak{G}$ and $\mathfrak{G}'$ be sequents $\mathcal{R}, \Gamma \Longrightarrow \Delta$ and $\mathcal{R}', \Gamma' \Longrightarrow \Delta'$ respectively. We define their ***sum*** $\mathfrak{G} + \mathfrak{G}'$ to be the sequent $\mathcal{R}, \mathcal{R}', \Gamma, \Gamma' \Longrightarrow \Delta, \Delta'$.

Construction 5.2.8 (Layer Lifting I)**.** Let $\mathfrak{G}$ be a X-saturated sequent, L be a layer in $\mathfrak{G}$, and let x be a label in L. Let $L = \{x, y_1, \ldots, y_l\}$ where $l \geq 0$. Let $\hat{L}$ be the set of fresh labels $\{\hat{x}, \hat{y}_1, \ldots, \hat{y}_l\}$. We define $\mathfrak{G}{\uparrow^x}$ to consist of:

1. relational atoms $v{\leq}v$ for all $v \in \hat{L}$;
2. for each $i = 1 \ldots l$, and every label w occurring in $\mathfrak{G}$:
   (a) relational atom $w{\leq}\hat{y}_i$ whenever $w{\leq_{\mathfrak{G}}}y_i$,
   (b) relational atom $w{\leq}\hat{x}$ whenever $w{\leq_{\mathfrak{G}}}x$;
3. for all $i, i' = 1..l$,
   (a) relational atom $\hat{y}_i R \hat{y}_{i'}$ whenever $y_i R_{\mathfrak{G}} y_{i'}$,
   (b) relational atom $\hat{y}_i R \hat{x}$ whenever $y_i R_{\mathfrak{G}} x$,
   (c) relational atom $\hat{x} R \hat{y}_i$ whenever $x R_{\mathfrak{G}} y_i$,
4. For every $i = 1..l$ and for every formula C, add:
   (a) labelled formulas $\hat{y}_i{:}C^\bullet$ whenever $\mathfrak{G}$,$y_i{:}C^\bullet$,
   (b) labelled formulas $\hat{x}{:}C^\bullet$ whenever $\mathfrak{G}$,$x{:}C^\bullet$.

This construction lifts a layer L of $\mathfrak{G}$ with respect to a label x in L. In Construction 5.2.8, Points 1)–2) ensure that in sequent $\mathfrak{G} + \mathfrak{G}{\uparrow^x}$, the new layer is indeed above L as intended and ($\leq$tr), ($\leq$rf) are satisfied; Point 3) ensures (F_1), and (F_2); Point 4) ensures ($\mathsf{mon}^\bullet$). In particular, all the other conditions for X-structural saturation are preserved and $\mathfrak{G} + \mathfrak{G}{\uparrow^x}$ is tree-layered.

Construction 5.2.9 (Layer Lifting II)**.** Let $\mathfrak{G}$ be a X-saturated sequent with $\mathfrak{G}$,$x{:}F^\circ$ being X-unhappy for some label x and some formula F of shape $A \supset B$ or $\Box B$. Let L be the layer of x. We define $\mathfrak{G}{\uparrow^{x:F}}$ as follows:

- If $F = A \supset B$, then we define $\mathfrak{G}{\uparrow^{x:A\supset B}}$ to be the sequent $\mathfrak{G}{\uparrow^x}$ to which we add labelled formulas $\hat{x}{:}A^\bullet$ and $\hat{x}{:}B^\circ$ and we call $\hat{x}$ a ***suricata label of*** x (where $\hat{x}$ is as in Construction 5.2.8);
- If $A = \Box B$, then we define $\mathfrak{G}{\uparrow^{x:\Box B}}$ to be the sequent $\mathfrak{G}{\uparrow^x}$ to which we add $z{\leq}z$ and $z{:}B^\circ$ for a fresh label z, and additionally we add relational atoms vRz whenever $v \in \hat{L}$ and $vR\hat{x}$ is in $\mathfrak{G}{\uparrow^x}$. We then close under X-structural saturation. Here z is called a ***suricata label of*** x (where z is as in Construction 5.2.8).

Informally speaking, the sequent $\mathfrak{G} + \mathfrak{G}{\uparrow}^{x:F}$ contains precisely the relational atoms and labelled formulas that need to be added to $\mathfrak{G} + \mathfrak{G}{\uparrow}^{x}$, such that (i) the X-unhappy $\mathfrak{G}$,x:F° becomes X-happy (the suricata label contains the white formula responsible for this happiness), and (ii) the result is still X-structurally saturated and tree-layered. We will add one such layer for every X-unhappy $A \supset B^\circ$ and $\Box A^\circ$ in the sequent, making all of them at some point X-happy. For each layer that it is added, we calculate the saturation in order to make all the lifted layers X-almost happy.

If we do this addition of layers naively, we will not terminate. Therefore we need a loop-check. In the logics where transitivity does not appear, we are going to look for a repetition of the sequent in the future branch. To define formally this loop-check, we now define the notion of simulation between layers.

Definition 5.2.10 (Equivalent labels)**.** Let x and y be labels occurring in sequents $\mathfrak{G}$ and $\mathfrak{H}$ respectively. We say that x and y are ***equivalent***, in symbols $x \sim y$, iff for all formulas A, we have $\mathfrak{G}$,x:$A^\bullet$ iff $\mathfrak{H}$,y:$A^\bullet$ and also $\mathfrak{G}$,x:A° iff $\mathfrak{H}$,y:A°.

Definition 5.2.11 (Simulation of layer)**.** Let L' and L be layers in a layered sequent $\mathfrak{G}$. A ***layer simulation*** between L' and L is a non-empty binary relation $\mathsf{S} \subseteq (L' \times L) \cap \sim$ such that for all $x' \in L'$, $x, y \in L$,

($\mathsf{S1}$) whenever $x'\mathsf{S}x$ and $xR_{\mathfrak{G}}y$ then there exists $y' \in L'$ such that $x'R_{\mathfrak{G}}y'$ and $y'\mathsf{S}y$, and

($\mathsf{S2}$) whenever $x'\mathsf{S}x$ and $yR_{\mathfrak{G}}x$ then there exists $y' \in L'$ such that $y'R_{\mathfrak{G}}x'$ and $y'\mathsf{S}y$.

We say L' ***simulates*** L iff there is a layer simulation between L' and L.

Definition 5.2.12. Let L be a layer in a layered sequent $\mathfrak{G}$. We say L ***is simulated*** if and only if there is a layer L' in $\mathfrak{G}$, such that $L' < L$ and L' simulates L.

Definition 5.2.13 (Allowed formula)**.** Let $\mathfrak{G}$ be a X-saturated sequent with $\mathfrak{G}$,x:F°. Let L be the layer of x. We say that the formula x:F° is ***allowed*** iff:

- F is of the shape $A \supset B$ or $\Box B$; and,
- x:F° is X-unhappy; and,
- L is not simulated; and,
- for all layers L' if $L' < L$, then L' is X-happy.

5.2.4 Proof search algorithm

We have now all the ingredients for our proof/countermodel search algorithm, which is presented in Figure 5.2[3]. This algorithm, as it was mentioned before, terminates for the logics $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$ and we use the logic X from here to the end of the chapter as one of those logics.

The algorithm produces a sequence $\boldsymbol{\mathfrak{G}}_0, \boldsymbol{\mathfrak{G}}_1, \boldsymbol{\mathfrak{G}}_2, \ldots$ of sets of X-saturated sequents using the transformations discussed in this section. We terminate at Step 2 if all sequents in $\boldsymbol{\mathfrak{G}}_i$ are axiomatic, and in that case we can produce a proof in $\mathsf{labX}'_{\leq}$ (see Section 5.4); or we can terminate if there are no more allowed formulas, in that case we can construct a countermodel (see Section 5.3).

[3]We will explain the things marked in red in Chapter 6

0. Given a formula F, define $\mathfrak{G}_0(F)$ to be the sequent $r{\leq}r \implies r{:}F$ and let $\boldsymbol{\mathfrak{G}}'_0 := \{\mathfrak{G}_0(F)\}$.

1. For the set $\boldsymbol{\mathfrak{G}}'_i$, calculate a saturation $\boldsymbol{\mathfrak{G}}_i$.

2. If all sequents in $\boldsymbol{\mathfrak{G}}_i$ are axiomatic, then terminate.
The formula F is provable and we can give a proof of $\implies r{:}F$ in $\mathsf{labX}'_{\leq}$.

3. Otherwise, pick a non-axiomatic sequent $\mathfrak{G}_i \in \boldsymbol{\mathfrak{G}}_i$.

   (a) Pick an allowed formula $x{:}F^{\circ}$, compute $\mathfrak{G}_i{\uparrow}^{x:F^{\circ}}$ and set $\boldsymbol{\mathfrak{G}}'_{i+1} = (\boldsymbol{\mathfrak{G}}'_i \setminus \{\mathfrak{G}_i\}) \cup \{\mathfrak{G}_i + \mathfrak{G}_i{\uparrow}^{x:F^{\circ}}\}$ and go to Step 1.

   (b) If we do not have any more allowed formulas, then terminate.
The formula F is not provable, and the sequent $\mathfrak{G}_i$ defines a countermodel (see Section 5.3).

Figure 5.2: Proof search algorithm

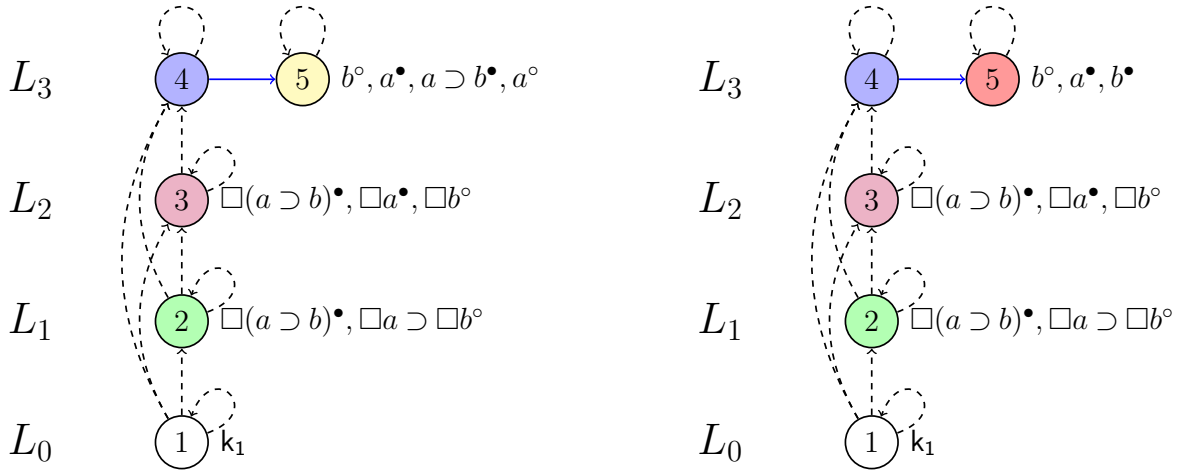


Figure 5.3: Axiom $\mathsf{k}_1 : \Box(a \supset b) \supset (\Box a \supset \Box b)$

Example 5.2.1 (Valid formula in IK)**.** Let us consider the axiom $\mathsf{k}_1 : \Box(a \supset b) \supset (\Box a \supset \Box b)$ which is a valid formula in IK. Figure 5.3 represents the sequents generated by the algorithm in Figure 5.2. Let $\Gamma^{\bullet}$ be the set $\{\Box(a \supset b)^{\bullet}, \Box a^{\bullet}\}$. To each label in the figure (except 1) we associate $\Gamma^{\bullet}$, plus the formulas explicitly displayed next to the node in the figure.

The algorithm starts in Step 0 and the next first step executed is Step 3, since the formula is non-axiomatic and there is an allowed formula: $1{:}\Box(a \supset b) \supset (\Box a \supset \Box b)^{\circ}$. This means we apply its lifting saturation (Definition 5.2.9) creating layer L_1 and label 2 where we add $\Box(a \supset b)^{\bullet}$ and $\Box a \supset \Box b^{\circ}$ and we go to Step 1. We calculate saturation, which in this case means to apply $\leq\mathsf{rf}$-structural saturation since the other formulas are already IK-happy. We go to Step 3 and we apply lifting saturation to the allowed formula $2{:}\Box a \supset \Box b^{\circ}$ creating layer L_2 and label 3 where we add $\Box a^{\bullet}$ and $\Box b^{\circ}$. We saturate L_2, in particular $(\leq\mathsf{rf})$-,

($\leq$tr)-, (mon$^\bullet$)-structural saturation (the $\Box^\bullet$-formulas are IK-happy). We go to Step 3 and we find the allowed formula $\Box b^\circ$, we then apply the lifting saturation creating layer L_3 and the labels 4 and 5 and we add in label 5 the formula b°. We go to Step 1 and calculate the saturation of L_3: we make IK-happy the $\Box^\bullet$-formulas, we structurally saturate the layer and we also make the formula $a \supset b^\bullet$ IK-happy in label 5. In order to make the $\supset^\bullet$ IK-happy, we have to create two different branches. This is the reason why there are two figures. In label 5 on the left figure we add a° and on the right we add $b^\bullet$. After saturating the layer, we go to Step 2 where we find that both sequents are axiomatic and the algorithm stops and according to Theorem 5.5.3, the formula at the root is a theorem of IK. Indeed, its derivation can be found in Example 5.4.1.

5.3 Countermodel construction

Let $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$. Assume we initiate the algorithm with a formula F. If we terminate at Step 3.b) in Figure 5.2, we have found a X-saturated sequent $\mathfrak{G}_i$ such that $\mathfrak{G}_{\mathfrak{i}}$,$r$:$F^\circ$ and there is no more allowed formulas. This means that each layer L of $\mathfrak{G}_i$ either is already X-happy or is simulated. In other words, if we terminate at Step 3.b), there are two possible cases: i) the non-axiomatic sequent $\mathfrak{G}_i$ does not have any X-unhappy formulas $\mathfrak{G}_{\mathfrak{i}}$,$x$:$F^\circ$, and therefore we terminate; ii) there are X-unhappy $\mathfrak{G}_{\mathfrak{i}}$,$x$:$\Box A^\circ$ or $\mathfrak{G}_{\mathfrak{i}}$,$x$:$A \supset B^\circ$, and we construct a X-happy sequent $\mathfrak{G}_i^*$. For each X-unhappy layer L there is some X-happy layer L' such that $L' < L$ and L' simulates L via simulation S_L (see Definition 5.2.11). We define $\mathfrak{G}_i^*$ to be obtained from $\mathfrak{G}_i$ by adding a relational atom $x \leq x'$ whenever $x \mathsf{S}_L x'$ for some X-unhappy layer L, and by closing the result under transitivity of $\leq$.

Then all the X-unhappy $\mathfrak{G}_{\mathfrak{i}}$,$x$:$\Box A^\circ$ and $\mathfrak{G}_{\mathfrak{i}}$,$x$:$A \supset B^\circ$ become X-happy since we create a new future x' for x where the conditions for happiness of $\mathfrak{G}_{\mathfrak{i}}$,$x$:$\Box A^\circ$ and $\mathfrak{G}$,x:$A \supset B^\circ$ are satisfied. Therefore, we have that the X-unhappy formulas in x are now X-happy. Hence, $\mathfrak{G}_i^*$ is a X-happy sequent, which allow us to apply Theorem 4.3.1 to obtain a finite countermodel for F.

Theorem 5.3.1. *If the algorithm shown in Figure 5.2 terminates in Step 3.b), then the formula F is not a theorem of* X.

Proof. If the algorithm terminates in Step 3.b), then there are no more allowed formulas in some non-axiomatic sequent $\mathfrak{G}_i \in \boldsymbol{\mathfrak{G}}_i$. It is easy to see that label r, the only label in $\boldsymbol{\mathfrak{G}}_0$ is never removed by the algorithm, since saturation and lifting saturation do not remove labels. Hence, $\mathfrak{G}_{\mathfrak{i}}$,$r$:$F^\circ$.

For any such X-unhappy $\mathfrak{G}_{\mathfrak{i}}$,$x$:$A^\circ$ from a layer L of $\mathfrak{G}_i$, there is a layer L' such that $L' < L$ and L' simulating L (see Definition 5.2.11) via a simulation S_L. Let $\mathfrak{G}$ be obtained by first adding to $\mathfrak{G}_{\mathfrak{i}}$ all relational atoms $x \leq_{\mathfrak{G}} x'$ such that $x' \mathsf{S}_L x$ and then closing the result under transitivity of $\leq$. At the same time $x \sim x'$ so $\mathfrak{G}_{\mathfrak{i}}$,$x'$:$A^\circ$ is X-happy in L'. If $A^\circ = B \supset C^\circ$, we must have $\mathfrak{G}_{\mathfrak{i}}$,$z$:$B^\bullet$, $\mathfrak{G}_{\mathfrak{i}}$,$z$:$C^\circ$ and $x' \leq_{\mathfrak{G}} z$ for some label z. Then we get $x \leq_{\mathfrak{G}} z$ by ($\leq$tr)-structural saturation of $\mathfrak{G}$, which makes $\mathfrak{G}$,x:A° to be X-happy. Finally, if $A^\circ = \Box B^\circ$, we must have $\mathfrak{G}_{\mathfrak{i}}$,$z$:$B^\circ$, $x' \leq_{\mathfrak{G}} u$ and $u R_{\mathfrak{G}} z$ for some labels u, z. Then we get $x \leq_{\mathfrak{G}} u$ by ($\leq$tr)-structural saturation of $\mathfrak{G}$ and uRz by construction, which makes $\mathfrak{G}$,x:A° X-happy.

We need to show that $\mathfrak{G}$ is X-structurally saturated and all its formulas are X-happy. Since no new labels were added, ($\leq$rf)-structural saturation is preserved. ($\leq$tr)-structural saturation is explicitly enforced. To show that (mon$^\bullet$)-, (F_1)-, and (F_2)-structural saturation

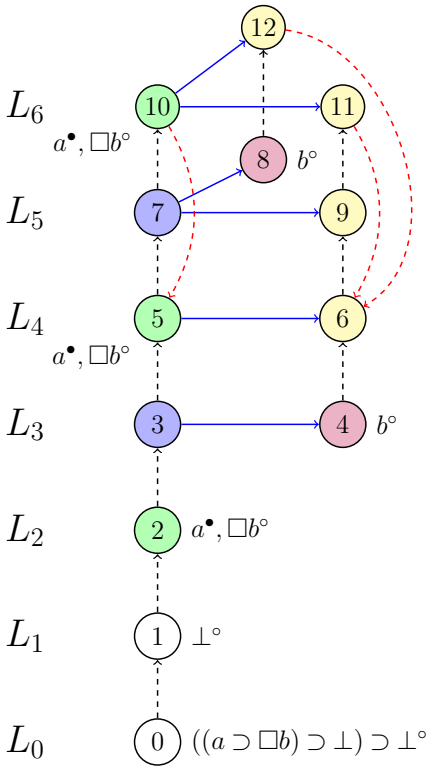


Figure 5.4: Non-valid formula in IK: $((a \supset \Box b) \supset \bot) \supset \bot$

are preserved, it is sufficient to demonstrate them for each of the $\leq$-links added before the transitive closure. For the (F_1)-, and (F_2)-structural saturation these are exactly the simulation conditions S1 and S2 from Definition 5.2.11. For the ($\mathsf{mon}^\bullet$)-simulation this follows from the fact that $\mathsf{S}_L \subseteq \sim$ for all L, which means that whenever $\mathfrak{G}, x{:}A^\bullet$ and $x \leq_{\mathfrak{G}} x'$ because $x' \mathsf{S}_L x$ for some L, we have $x \sim x'$ and, hence, $\mathfrak{G}, x'{:}A^\bullet$. Other conditions of X-structural saturation depending on the logic X are preserved because no new R-links were added. This completes the proof that $\mathfrak{G}$ is X-structurally saturated. Since no formulas or links were removed, all formulas X-happy in $\mathfrak{G}_{\mathfrak{i}}$ remain X-happy in $\mathfrak{G}$. This completes the proof that $\mathfrak{G}$ is a X-happy sequent. Then we can apply 5.1.7 and we get $\mathcal{M}_{\mathfrak{G}}, r \not\Vdash F$ and, by Theorem 3.2.1, F is not a theorem in IS4. □

Example 5.3.1 (Non-valid formula in IK)**.** Figure 5.4 represents a $\leq$-branch of a sequent, generated by the algorithm when run on formula $((a \supset \Box b) \supset \bot) \supset \bot$. Let $\Gamma^\bullet = \{\bot \supset (a \supset \Box b)^\bullet, a \supset \Box b^\circ\}$. To each label in the figure (except 0), we associate $\Gamma^\bullet$, plus the formulas explicitly displayed next to the node.

The search on the depicted $\leq$-branch stops because the layer L_6 can be simulated by layer L_4. By adding relational atoms $10 \leq 5$, $11 \leq 6$ and $12 \leq 6$ to the sequent (by Theorem 5.3.1), which are the dashed red arrows pointing downwards in the figure, we obtain (a part of) the countermodel for our formula. To complete the countermodel, we need to take into account the layers generated from other IK-unhappy $\Box^\circ$ and $\supset^\circ$ formulas present in the sequent.

Remark 5.3.2. Observe that the first attempt would be to try to look for equality between an upper layer and a lower layer of the sequent. However, this is not possible due to the

$$\mathsf{mon}^\bullet\ \frac{\mathcal{R}, x{\leq}y, \Gamma, x{:}A, y{:}A \Longrightarrow \Delta}{\mathcal{R}, x{\leq}y, \Gamma, x{:}A \Longrightarrow \Delta} \qquad \mathsf{cont}^\bullet\ \frac{\mathcal{R}, \Gamma, x{:}A, x{:}A \Longrightarrow \Delta}{\mathcal{R}, \Gamma, x{:}A \Longrightarrow \Delta}$$

$$\mathsf{weak}\ \frac{\mathcal{R}, \Gamma \Longrightarrow \Delta}{\mathcal{R}, \mathcal{R}', \Gamma, \Gamma' \Longrightarrow \Delta, \Delta'} \qquad \mathsf{cont}^\circ\ \frac{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A, x{:}A}{\mathcal{R}, \Gamma \Longrightarrow \Delta, x{:}A}$$

Figure 5.5: Admissible rules in $\mathsf{labX}_{\leq}$

interaction between $\Box^\circ$ and F_1 and F_2. The rule $\Box^\circ$ is repeatedly creating new branches as it is shown in Figure 5.4.

5.4 Proof construction

Let us now turn to the case when the algorithm terminates in Step 2. Then all sequents in $\mathcal{G}_i$ are axiomatic, and we want to construct a proof of $\Longrightarrow r{:}F$ in $\mathsf{labX}_{\leq}$ for $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$. For this we are going to simulate the steps of the algorithm by applying the inference rules of $\mathsf{labX}'_{\leq}$ which is define as $\mathsf{labX}'_{\leq} = \mathsf{labX}_{\leq} \cup \{\mathsf{mon}^\bullet, \mathsf{weak}, \mathsf{cont}^\bullet, \mathsf{cont}^\circ\}$, where the rules $\mathsf{mon}^\bullet, \mathsf{weak}, \mathsf{cont}^\bullet, \mathsf{cont}^\circ$ are admissible (see Proposition 4.2.3, Lemma 4.5.2 and Remark 4.2.4 in Chapter 4) and they are also presented in Figure 5.5.

We start with the rule $\leq\mathsf{rf}$ to obtain the sequent $r{\leq}r \Longrightarrow r{:}F$ that is the input of the algorithm in Step 0. Now each step of the algorithm can be executed by applying the rules of $\mathsf{labX}'_{\leq}$. Step 1 corresponds to the application of all the inference rules that do not add new labels in the future branch. In other words, we start the derivation by applying all the rules of $\mathsf{labX}'_{\leq}$ except for the rules $\Box^\circ$ and $\supset^\circ$. In the case we can only apply $\Box^\circ$ or $\supset^\circ$; or if we do not obtain a proof by only applying the other inference rules, we then continue with the application of the rules $\Box^\circ$ or $\supset^\circ$, and rules $\mathsf{mon}^\bullet, \mathsf{F}_1, \mathsf{F}_2, \leq\mathsf{tr}, \leq\mathsf{rf}$ (this is happening in the Step 3 of the algorithm in the lifting saturation, which involves the lifting of the whole layer as Construction 5.2.8 and the lifting coming from $\Box^\circ$ and $\supset^\circ$ as Construction 5.2.9). We repeat this procedure until we can apply the rule id and get a proof for our formula F. Then F is a theorem of the logic X.

Example 5.4.1. We consider the valid formula for the logic IK presented in Example 5.2.1 and we obtain its proof tree in $\mathsf{labIK}'_{\leq}$ below. Observe that the first rule we can apply is $\supset^\circ$ which corresponds to the execution of Step 3 in Example 5.2.1 where L_1 is created. Our algorithm then saturates the layer which corresponds to the application of all the rules which are not creating new labels (Step 1): this means in our proof construction, we apply $\leq\mathsf{rf}$ (there is nothing else to apply at this step and, in particular, we cannot apply $2{:}\Box(A \supset B)^\bullet$ since it needs an accessibility relation, i.e. according to the algorithm, it is already IK-happy). We then continue applying the next $\supset^\circ$-rule (as before, this is Step 3 in the algorithm and $2{:}\Box A \supset \Box B$ is an allowed formula). This rule introduces the fresh label 3 and it adds the rules $3{:}\Box A^\bullet$ and $3{:}\Box B^\circ$. We now apply again the saturation rules: in this case we apply $\leq\mathsf{tr}$, $\leq\mathsf{rf}$ and $\mathsf{mon}^\bullet$ (there is nothing else to do in the saturation Step 2 in the algorithm). We continue applying $\Box^\circ$ which is adding labels 4 and 5 (creating layer L_3 in Example 5.2.1) and adding the formula $5{:}B^\circ$ (in Step 3 of the algorithm, $3{:}\Box B^\circ$ is an allowed formula and we apply its lifting as Construction 5.2.9). We apply again the saturation rules: $\leq\mathsf{rf}, \leq\mathsf{tr}, \mathsf{mon}^\bullet, \Box^\bullet$(two times) and $\supset^\bullet$ which is creating new branches. In

each branch we can apply the rule $\mathsf{id_g}$ and finish our proof. In the algorithm, this last step is Step 2 in which all the sequents are axiomatic and we terminate. The two figures of Figure 5.3 represent the leaves of this proof tree, left and right respectively.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{R}, 3{:}\Box(A \supset B), 3{:}\Box A, 4{:}\Box A, 4{:}\Box(A \supset B), 5{:}A, 5{:}A \supset B \Longrightarrow 5{:}B, 5{:}A}\,\mathsf{id_g}}{\mathcal{R}, 3{:}\Box(A \supset B), 3{:}\Box A, 4{:}\Box A, 4{:}\Box(A \supset B), 5{:}A, 5{:}A \supset B \Longrightarrow 5{:}B}\,\supset^\bullet \quad \dfrac{}{\mathcal{R}, 3{:}\Box(A \supset B), 3{:}\Box A, 4{:}\Box A, 4{:}\Box(A \supset B), 5{:}A, 5{:}B \Longrightarrow 5{:}B}\,\mathsf{id_g}}{\mathcal{R}, 3{:}\Box(A \supset B), 3{:}\Box A, 4{:}\Box A, 4{:}\Box(A \supset B), 5{:}A, 5{:}A \supset B \Longrightarrow 5{:}B}\,\supset^\bullet}{\mathcal{R}, 3{:}\Box(A \supset B), 3{:}\Box A, 4{:}\Box A, 4{:}\Box(A \supset B), 5{:}A \Longrightarrow 5{:}B}\,\Box^\bullet}{\mathcal{R}, 3{:}\Box(A \supset B), 3{:}\Box A, 4{:}\Box A, 4{:}\Box(A \supset B) \Longrightarrow 5{:}B}\,\Box^\bullet}{1{\leq}1, 1{\leq}2, 2{\leq}2, 2{\leq}3, 3{\leq}3, 1{\leq}3, 3{\leq}4, 4R5, 2{:}\Box(A \supset B), 3{:}\Box(A \supset B), 3{:}\Box A \Longrightarrow 5{:}B}\,\leq\mathsf{rf}{+}\leq\mathsf{tr}{+}\mathsf{mon}^\bullet$$

$$\dfrac{1{\leq}1, 1{\leq}2, 2{\leq}2, 2{\leq}3, 3{\leq}3, 1{\leq}3, 2{:}\Box(A \supset B), 3{:}\Box(A \supset B), 3{:}\Box A \Longrightarrow 3{:}\Box B}{1{\leq}1, 1{\leq}2, 2{\leq}2, 2{\leq}3, 2{:}\Box(A \supset B), 3{:}\Box A \Longrightarrow 3{:}\Box B}\,\leq\mathsf{rf}{+}\supset^\circ\ 3 \text{ fresh}$$

$$\dfrac{1{\leq}1, 1{\leq}2, 2{\leq}2, 2{:}\Box(A \supset B) \Longrightarrow 2{:}\Box A \supset \Box B}{1{\leq}1 \Longrightarrow 1{:}\Box(A \supset B) \supset (\Box A \supset \Box B)}\,\supset^\circ\ 2 \text{ fresh}$$

$$\dfrac{1{\leq}1, 1{\leq}2, 2{\leq}2, 2{\leq}3, 3{\leq}3, 1{\leq}3, 2{:}\Box(A \supset B), 3{:}\Box(A \supset B), 3{:}\Box A \Longrightarrow 3{:}\Box B}{1{\leq}1, 1{\leq}2, 2{\leq}2, 2{\leq}3, 3{\leq}3, 1{\leq}3, 3{\leq}4, 4R5, 2{:}\Box(A \supset B), 3{:}\Box(A \supset B), 3{:}\Box A \Longrightarrow 5{:}B}\,\leq\mathsf{rf}{+}\leq\mathsf{tr}{+}\mathsf{mon}^\bullet$$

$$\dfrac{1{\leq}1, 1{\leq}2, 2{\leq}2, 2{\leq}3, 3{\leq}3, 1{\leq}3, 3{\leq}4, 4R5, 2{:}\Box(A \supset B), 3{:}\Box(A \supset B), 3{:}\Box A \Longrightarrow 5{:}B}{\mathcal{R}, 3{:}\Box(A \supset B), 3{:}\Box A, 4{:}\Box A, 4{:}\Box(A \supset B) \Longrightarrow 5{:}B}\,\Box^\circ\ 4, 5 \text{ fresh}$$

where $\mathcal{R}$ is equal to: $1{\leq}1, 1{\leq}2, 2{\leq}2, 2{\leq}3, 3{\leq}3, 1{\leq}3, 3{\leq}4, 1{\leq}4, 4{\leq}4, 2{\leq}4, 5{\leq}5, 4R5, 2{:}\Box(A \supset B)$.

5.5 Termination

We have already established in Section 5.2 that every step in our algorithm (shown in Figure 5.2) terminates for the logic $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$. It remains to show that we cannot run through the main loop forever, i.e., we do not produce an infinite sequence $\mathfrak{G}_0, \mathfrak{G}_1, \ldots, \mathfrak{G}_i, \ldots$ of sets of sequents but eventually terminate either in Step 2 (we find a proof) or in Step 3.b) (we find a countermodel). The basic idea is to restrict the size of a layer in the sequents of a set $\mathfrak{G}_i$. This then means that the number of distinct branches in a layer is finite, hence, we will eventually find a simulation (Definitions 5.2.11 and 5.2.12).

For the remainder of this section, we assume we have started the algorithm with a formula F in Step 0, and that n is the number of subformula occurrences in F.

We define the ***size*** of a label x in a sequent $\mathfrak{G}$, denoted as $|x|$, to be the number of distinct formula occurrences $x{:}A^\bullet$ or $x{:}A^\circ$ in $\mathfrak{G}$.

Lemma 5.5.1. *The size of a label occurring in a sequent of some $\mathfrak{G}_i$ is at most n. And there are 2^n many equivalence classes of labels with respect to $\sim$.*

Proof. Only subformulas of F can occur in a sequent. Furthermore, the position of each subformula occurrence in F determines if such a subformula can occur on the left or on the right of the $\Longrightarrow$. Moreover, the algorithm introduces in a sequent only labelled formulas which do not already occur in the sequent (Definition 5.2.1). Hence, we can have at most n formula occurrences at a given label. Also note that $x \sim y$ if and only if x and y contain the same set of subformulas of F. Hence, there are 2^n different equivalence classes for $\sim$ on labels which can occur in $\mathfrak{G}_i$. □

Lemma 5.5.2. *The length of a branch in a layer in a sequent in a set $\mathfrak{G}_i$ is bounded, and the bound is determined by F.*

Proof. A branch in a layer is getting longer by making the $\Diamond^\bullet$-formulas X-happy, since this rule is creating new labels in the same R-branch of a layer. We have already shown in Lemma 5.2.6 that we will not keep adding new labels forever. Observe that all the structural rules of every logic $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$ are not moving $\Diamond^\bullet$ to other labels, i.e. they

are not propagating $\Diamond^\bullet$ and, therefore, the modal degree can only decreased. Therefore, the length of a branch in a layer is bounded, and it is bound by the number of $\Box$ and $\Diamond$ in the formula F. $\Box$

Theorem 5.5.3. *The proof search algorithm given in Figure 5.2 is terminating.*

Proof. By the previous lemma, the size of a branch in a layer is bounded by the formula F in the end sequent. Eventually there will be a simulation by a previous layer in the algorithm, and therefore, the number of possible layers that can occur in a sequent in a set $\mathfrak{G}_i$ that is visited by the algorithm is also bounded. This puts a limit to the height of the tree of layers in the sequents. $\Box$

5.6 Conclusion

In this chapter we have presented a proof search algorithm for intuitionistic modal logics in the $\mathsf{S5}$-cube. We show that this algorithm does terminate for some logics where transitivity is not one of the conditions restricting the class of frames. In particular, it is indeed terminating for the logics $\mathsf{X} \in \{\mathsf{IK}, \mathsf{IT}, \mathsf{IKB}, \mathsf{ITB}, \mathsf{IK5}\}$.

Our solution uses our fully labelled sequent system $\mathsf{labX}_{\leq}$ with relational atoms for both binary relations (accesibility relation R and preorder relation $\leq$) which enabled us to give a proof system that has only invertible rules and also gives a closer correspondence between sequents and models. This is a key ingredient for the decision problem for intuitionistic modal logic $\mathsf{IS4}$ which decidability was an open problem until now [21]. In fact, we will apply the same argument presented in this chapter, but in many definitions and proof arguments, there would be subtle differences due to the presence of transitivity.

Chapter 6

Decision problem for IS4

We have presented a decision procedure for intuitionistic modal logics of the S5-cube which it is terminating only for a few logics (none of them including transitivity). In this chapter we show how to ensure termination for the intuitionistic modal logic IS4. In particular, this solves a problem that has been open for almost thirty years since it had been posted in Simpson's PhD thesis in 1994 [59]. This is a joint work with Girlando, Kuznets, Marin and Straßburger [21]. Our search algorithm outputs either a proof or a finite countermodel, thus, additionally establishing the finite model property for IS4, which has been another long-standing open problem in the area. We obtained these results by using the general argument of the algorithm presented in Chapter 5 and by adding the corresponding new notions and loop-checks that are needed.

We start this chapter with Section 6.1 where we highlight the main difficulties that we encountered in tackling the decidability problem for IS4. We then follow with the same structure as Chapter 5. In Section 6.2, we present the search algorithm to prove decidability of IS4 and the necessary new notions in order to ensure its termination. In Section 6.3 we show how to obtain a countermodel from a failed proof search. In Section 6.4 we show how to reconstruct a real proof when the algorithm succeeds. In Section 6.5 we prove that the algorithm always terminates. We end the chapter with Section 6.6 where we present our conclusions.

6.1 Why is it a hard problem?

As it was mentioned before, one way to prove decidability for a logic is to perform proof search in a sound and complete deductive system with the intention of either finding a proof or constructing a countermodel from a failed proof search. For IS4 (it is obtained from IK by adding the t and 4 axioms) several such deductive systems exist, the first being Simpson's labelled systems presented in his PhD thesis [59] and already introduced in Chapter 3. Moreover, there are various kinds of nested sequents systems: single-conclusion [60], multiple conclusion [61], and also focused variants [10]. A natural question to ask is why none of these systems has been used to prove decidability of IS4. The aforementioned systems rely on what we could call a mixed approach: they internalize the modal accessibility relation R within the sequents syntax, either using nesting, or labels and relational atoms, but they rely on a traditional structural approach for the intuitionistic aspect of the logic such as single-conclusion sequents at least in certain rules.

One might think that combining the traditional loop-check for the intuitionistic part with the label-based loop-check for the modal part would be a way to a decision procedure. But the situation is more complicated because

1. all sequent systems mentioned above internalized only the R-relation of the birelational models. If the system is also single-conclusion (as most intuitionistic systems are), then the classical $\mathsf{S4}$-loop test, which is looking for a repetition along the R-relation, cannot be applied to the right-hand-side of a sequent, as the conclusion formula sometimes can be replaced by a new one;

2. the structural approach to the intuitionistic system also means the rules are not all invertible and the procedure requires backtracking, so the modality loop check also needs to be combined with the necessary backtracking.

These two problems can be overcome by using our fully labelled proof system that incorporates both relations R and $\leq$ [43]. This has the same advantages as moving from a structural to a labelled approach for intuitionistic propositional logic as it was mentioned in the previous Chapter. Not only does this system re-establish the close relationship between a sequent and a model, as we know it for classical modal logic, it also enables us to make all rules in the system invertible. Moreover, by having access to explicit relational atoms in the sequent syntax, it makes it easy to implement the loop checks and to represent the backedges explicitly when constructing a countermodel as it was done in the previous chapter.

Naive proof search is not terminating, with two possible sources of non-termination: the first inherited from the classical modal logic $\mathsf{S4}$, since sequents can grow arbitrarily because the rules $\Diamond^\bullet$ and $\Box^\circ$ create new sequent nodes and the rules $\mathsf{4}^\bullet$ and $\mathsf{4}^\circ$ can move formulas into a child node without reducing their complexity. In classical $\mathsf{S4}$ this is prevented by a loop check: before a new child node is created by $\Diamond^\bullet$ or $\Box^\circ$, it is checked whether an ancestor sequent node is identical to the current one. If this is the case, the child node is not created. The second source of infinity comes from intuitionistic propositional logic. The rule $\supset^\circ$ removes the output formula from the sequent and introduces a new one. This can cause the algorithm to visit the same sequent again and again. To prevent this from happening the same loop check that we specifically used in the previous chapter can be attempted for use once more: check whether the same sequent has been visited before on the same branch of the proof search tree, and only if this is not the case, we continue the search. However, it is not straightforward to combine the two proof search methods for classical modal logic and propositional intuitionistic logic, as the following example shows:

Example 6.1.1. Consider the following formula, which is not provable in $\mathsf{IS4}$:

$$\Box\Big((\Box a \supset \bot) \wedge \Big((a \supset \bot) \supset \bot\Big)\Big) \supset \bot. \tag{6.1}$$

Let $A = \Box a \supset \bot$ and $B = (a \supset \bot) \supset \bot$. In order to construct a countermodel $\mathcal{M}$, we need a world w_1 that forces $\Box(A \wedge B)$ and, therefore, also forces A and forces B. Consequently, every world w' such that $w_1 \leq w$ and wRw' for some w should force these formulas. Then all these worlds must force neither $\Box a$ nor $a \supset \bot$. The latter means that for each such world w', there must be a world v with $w' \leq v$ and $\mathcal{M}, v \Vdash a$. Of course, in turn, v must not force $\Box a$, so there must be worlds u and u' with $v \leq u'$, $u'Ru$, and $\mathcal{M}, u \nVdash a$. But this u must not force $a \supset \bot$ so there must be a world v_1 with $u \leq v_1$ and $\mathcal{M}, v_1 \Vdash a$, and so on. Thus,

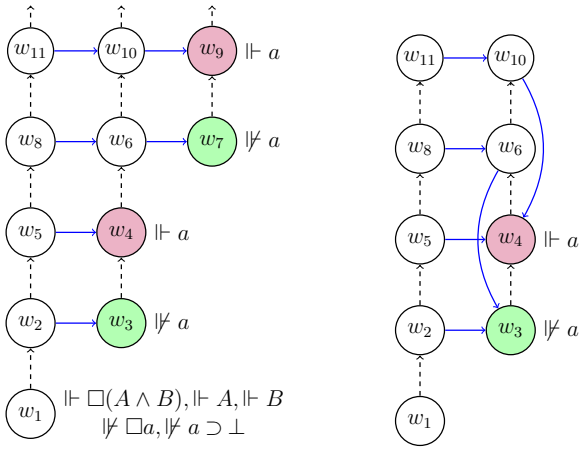


Figure 6.1: Left: Illustration of the potential non-termination issue. Right: Illustration of a break of condition $\mathsf{F_1}$ when identifying nodes unrestrictedly.

a naive implementation of a countermodel construction via proof search will keep adding worlds *ad infinitum* because neither of the two loop-checks will detect the repetition (see Figure 6.1, Left). In other words,

3. to account for the interaction of modalities and intuitionistic implications, the two loop checks must get along well with each other.

For solving this third problem, we have to implement a more sophisticated loop check involving both relations. This directly leads us to a fourth problem.

Example 6.1.2. Assume, for the sake of example, that we designed a suitable loop check such that we could stop proof search at the stage of the structure presented on Figure 6.1 (Left) and that we identified w_7 to w_3 and w_9 to w_4. This would create "backlinks" between w_{10} and w_4 as well as between w_6 and w_3. However this would lead to an incorrect $\mathsf{F_1}$ situation as now $w_{10}Rw_4 \leq w_6$ but there is no w' such that $w_{10} \leq w'Rw_6$. (see Figure 6.1, Right). This means that

4. the standard method of constructing a (finite) countermodel from a failed proof search, via identifying labels/worlds that create a loop, fails in the setting of birelational models. We break the $\mathsf{F_1}/\mathsf{F_2}$ conditions, which in turn would force us to add new worlds, which would mean we have to continue proof search.

We solve this problem by identifying (substituting) labels not only after we finished the proof search, but while still performing it. This preserves unprovability, but could, *a priori*, be unsound. This means that when terminating a branch on a non-axiomatic sequent, it is still possible to extract a countermodel from it. However, when reaching only axiomatic leaves, it remains to be shown that a sound proof can be obtained from the proof attempt (potentially containing identification of labels). So, instead of doing naive proof search and then constructing a countermodel from a failed proof search by "folding" the failed sequent, we perform the folding already during the proof search, which is now a countermodel search, and then construct a proper proof from a failed countermodel search by "unfolding" the search tree. The loop check ensuring termination has to be subtly calibrated for this final step of unfolding the proof attempt into a real proof.

6.2 Search algorithm for IS4

As it was mentioned in previous chapters, the intuitionistic modal logic IS4 is obtained by adding the axiom t for reflexivity and the axiom 4 for transitivity to the logic IK. In order to prove decidability of IS4, we use our fully labelled proof system $\mathsf{labX}_{\leq}$, presented in Chapter 4 as it was used for decidability of other logics in Chapter 5. In particular for IS4 we perform proof search in $\mathsf{labIS4}_{\leq} = \mathsf{labIK}_{\leq} \cup \{R\mathsf{tr}, R\mathsf{rf}\}$ (this two rules are presented in Figure 4.2) which is sound and complete by Theorem 4.6.2.

We will use most of the definitions presented in Chapter 5 for the logic X where now X = IS4. We work on IS4-structurally saturated sequents (see Definition 5.1.4) and making the formulas IS4-happy (see Definition 5.1.2). We are also using the same search algorithm presented in Section 5.2, with the necessary changes and adding the needed loop-checks in order to ensure termination.

In the same way as in Chapter 5, in all sequents the order relation $\leq$ on layers defines a tree structure. However, in sequents constructed by the proof system $\mathsf{labIS4}_{\leq}$, each layer also has a tree structure with respect to $R_{\mathfrak{G}}$. But, in order to search for a proof and a countermodel at the same time, we need to weaken this tree structure on the layers. For this, we introduce the notion of *clusters*.

Definition 6.2.1 (Cluster)**.** If $\mathfrak{G}$ is a IS4-structurally saturated sequent, then $R_{\mathfrak{G}} \cap R_{\mathfrak{G}}^{-1}$ is an equivalence relation, and we can define a ***cluster*** C in $\mathfrak{G}$ to be an equivalence class $C = \{x_1, ..., x_n\}$ of $R_{\mathfrak{G}} \cap R_{\mathfrak{G}}^{-1}$. A cluster $C = \{x_1\}$ containing only one label is called ***singleton***. On clusters, we define the following binary relations:

- $C_1 \leq_{\mathfrak{G}} C_2$ iff for all $y \in C_2$ there is $x \in C_1$ with $x \leq_{\mathfrak{G}} y$.
- $C_1 R_{\mathfrak{G}} C_2$ iff there are $x \in C_1$ and $y \in C_2$ with $x R_{\mathfrak{G}} y$.

For these two relations on clusters, we sometimes abuse the notation and replace one of the clusters by a label x even when $\{x\}$ is not a cluster. Nevertheless, the definitions are then applied verbatim to $\{x\}$.

Definition 6.2.2 (Equivalent clusters)**.** For IS4-structurally saturated sequents $\mathfrak{G}$ and $\mathfrak{H}$, we can generalize Definition 5.2.10 to clusters: $C_1 \sim C_2$ iff there is a bijection $f\colon C_1 \to C_2$, such that $fx \sim x$ for all $x \in C_1$.

Proposition 6.2.3. *For a* IS4*-structurally saturated sequent* $\mathfrak{G}$*,* $R_{\mathfrak{G}}$ *is an order and* $\leq_{\mathfrak{G}}$ *is a preorder on its clusters. If* $\mathfrak{G}$ *is layered,* $\leq_{\mathfrak{G}}$ *is also an order.*

Proof. $R_{\mathfrak{G}}$ is an order on its clusters:

- For reflexivity of $R_{\mathfrak{G}}$, consider any cluster C. For any label $x \in C$, by ($R\mathsf{rf}$)-structural saturation, we have $x R_{\mathfrak{G}} x$. Hence, $C R_{\mathfrak{G}} C$ by Definition 6.2.1.
- For transitivity of $R_{\mathfrak{G}}$, let $C_1 R_{\mathfrak{G}} C_2$ and $C_2 R_{\mathfrak{G}} C_3$ for clusters C_1, C_2, and C_3. By Definition 6.2.1, $x R_{\mathfrak{G}} y$ and $u R_{\mathfrak{G}} z$ for some labels $x \in C_1$, $y, u \in C_2$, and $z \in C_3$. Since $y R_{\mathfrak{G}} u$, by ($R\mathsf{tr}$)-structural saturation $x R_{\mathfrak{G}} z$. Thus, $C_1 R_{\mathfrak{G}} C_3$ by Definition 6.2.1.

- For antisymmetry of $R_{\mathfrak{G}}$, let $C_1 R_{\mathfrak{G}} C_2$ and $C_2 R_{\mathfrak{G}} C_1$. By Definition 6.2.1, there are labels $x, y \in C_1$ and $x', y' \in C_2$ such that $x R_{\mathfrak{G}} x'$ and $y' R_{\mathfrak{G}} y$. For arbitrary labels $u \in C_1$ and $v \in C_2$, we have $u R_{\mathfrak{G}} x$, and $x' R_{\mathfrak{G}} v$, hence, by ($R\mathsf{tr}$)-structural saturation, $u R_{\mathfrak{G}} v$. Similarly, $v R_{\mathfrak{G}} u$ because $v R_{\mathfrak{G}} y'$ and $y R_{\mathfrak{G}} u$. Since both $u R_{\mathfrak{G}} v$ and $v R_{\mathfrak{G}} u$ for all $u \in C_1$ and $v \in C_2$, all labels in these two clusters form one equivalence class w.r.t. $R_{\mathfrak{G}} \cap R_{\mathfrak{G}}^{-1}$, i.e., $C_1 = C_2$.

$\leq_{\mathfrak{G}}$ is a preorder on its clusters:

- For reflexivity of $\leq_{\mathfrak{G}}$, consider any cluster C. For every label $y \in C$, by ($\leq\mathsf{rf}$)-structural saturation, $y \leq_{\mathfrak{G}} y$. Hence, $C \leq_{\mathfrak{G}} C$ by Definition 6.2.1.

- For transitivity of $\leq_{\mathfrak{G}}$, let $C_1 \leq_{\mathfrak{G}} C_2$ and $C_2 \leq_{\mathfrak{G}} C_3$ for clusters C_1, C_2, and C_3. By Definition 6.2.1, for every $z \in C_3$, there is $y \in C_2$ such that $y \leq_{\mathfrak{G}} z$. In its turn, for this y, there is $x \in C_1$ such that $x \leq_{\mathfrak{G}} y$. By ($\leq\mathsf{tr}$)-structural saturation $x \leq_{\mathfrak{G}} z$ for this x. Thus, $C_1 \leq_{\mathfrak{G}} C_3$ by Definition 6.2.1. Hence, $\leq_{\mathfrak{G}}$ is a preorder.

Assume now additionally that $\mathfrak{G}$ is layered. Then to show that $\leq_{\mathfrak{G}}$ is also an order, it remains to show that $\leq_{\mathfrak{G}}$ is antisymmetric:

- For antisymmetry of $\leq_{\mathfrak{G}}$, let $C_1 \leq_{\mathfrak{G}} C_2$ and $C_1 \neq C_2$, i.e., $C_1 \cap C_2 = \varnothing$. To show that $C_2 \not\leq_{\mathfrak{G}} C_1$, we consider any label $x \in C_1$ and show that $y \not\leq_{\mathfrak{G}} x$ for all $y \in C_2$. By Definition 6.2.1, for each $y \in C_2$, there is some label $x' \in C_1$ such that $x' \leq_{\mathfrak{G}} y$. We have $x' R^{\leftrightarrow}_{\mathfrak{G}} x$ because they belong to the same cluster, $y R_{\mathfrak{G}} y$, and hence $y R^{\leftrightarrow}_{\mathfrak{G}} y$, by ($R\mathsf{rf}$)-structural saturation, and $x' \neq y$ because they are from disjoint clusters. Hence, $y \not\leq_{\mathfrak{G}} x$ by Definition 5.1.12. Since y was chosen arbitrarily, it follows that $C_2 \not\leq_{\mathfrak{G}} C_1$ and $\leq_{\mathfrak{G}}$ is an order. □

Definition 6.2.4 (Tree-clustered sequent)**.** A $\mathsf{IS4}$-structurally saturated sequent $\mathfrak{G}$ is called ***tree-clustered*** if and only if for all clusters C, C', and C'', whenever $C' R_{\mathfrak{G}} C$ and $C'' R_{\mathfrak{G}} C$, then either $C' R_{\mathfrak{G}} C''$ or $C'' R_{\mathfrak{G}} C'$.

Remark 6.2.5. Note that we do not need to ask for the root in Definition 6.2.4.

We now redefine the notions for saturated and semi-saturated sequents presented in Definition 5.1.18 in order to include the fact that the sequents are now tree-clustered.

Definition 6.2.6 ($\mathsf{IS4}$-saturated and $\mathsf{IS4}$-semi-saturated sequent)**.** A sequent $\mathfrak{G}$ of $\mathsf{IS4}$ is called $\mathsf{IS4}$-***saturated*** if and only if it is tree-layered and tree-clustered and $\mathsf{IS4}$-structurally saturated and all its formulas are $\mathsf{IS4}$-almost happy. A sequent $\mathfrak{G}$ is $\mathsf{IS4}$-***semi-saturated*** if and only if it is tree-layered and tree-clustered and $\mathsf{IS4}$-structurally saturated and all its formulas are $\mathsf{IS4}$-naively happy. A set $\boldsymbol{\mathfrak{G}}$ of sequents is called $\mathsf{IS4}$-***saturated/***$\mathsf{IS4}$-***semi-saturated*** if and only if it is finite and all elements of $\boldsymbol{\mathfrak{G}}$ are $\mathsf{IS4}$-saturated/$\mathsf{IS4}$-semi-saturated, respectively.

We are now ready to present the changes needed in the steps of the search algorithm of Figure 5.2 to ensure termination of $\mathsf{IS4}$. As it was done for the other logics X, we work on layers until these layers are saturated. This is achieved (as before) by semi-saturating the layer, i.e. applying all the inference rules that do not introduce new labels and then pick an $\mathsf{IS4}$-unhappy $x{:}\Diamond A^{\bullet}$ and make it $\mathsf{IS4}$-happy (either by applying the $\Diamond^{\bullet}$ rule or, in case of

loop detection, by creating a cluster). Then we semi-saturate again and we continue until all $\Diamond^\bullet$ in the layer are IS4-happy and, therefore, the layer is IS4-almost happy.

Now the only IS4-unhappy formulas remaining in the resulting IS4-saturated sequent are of the shape $x{:}\bot^\bullet, x{:}a^\circ, x{:}A \supset B^\circ, x{:}\Box A^\circ$. For the first two shapes, the sequent is axiomatic, and we can stop working on it as we have done in Chapter 5. And we continue looking for allowed formulas (which are of the shape $x{:}A \supset B^\circ$ or $x{:}\Box A^\circ$). We make them IS4-happy, until there are no more allowed formulas and we see a repetition of layers. In order to achieve termination of this process, we implement two other loop detection mechanisms (different from the one used for $x{:}\Diamond A^\bullet$ formulas).

In the remainder of this section, we formally introduce the changes for the algorithm including the mentioned loop checks.

6.2.1 Semi-saturation

We start with the semi-saturation procedure by making all formulas IS4-happy, that are not of the form $x{:}\bot^\bullet$, $x{:}a^\circ$, $x{:}\Diamond A^\bullet$, $x{:}A \supset B^\circ$ or $x{:}\Box A^\circ$. The process is the same as the one presented in Definition 5.2.1. However, since happiness for the logic IS4 is different for the formulas $x{:}\Box A^\bullet$ and $x{:}\Diamond A^\circ$ (see Definition 5.1.2), then the semi-saturation procedure needs to add the formulas $z{:}\Box A^\bullet$ in the case 3) and $y{:}\Diamond A^\circ$ in the case 4) of Definition 5.2.1.

We prove in Lemma 5.2.2, b) that the rewrite relation $\rightrightarrows_{\mathsf{s}}$ terminates for all the intuitionistic modal logics X in the S5-cube. In particular, it also terminates for IS4.

6.2.2 Saturation

In the next step we show how to deal with IS4-unhappy $x{:}\Diamond A^\bullet$ as we have done in the previous chapter. Nevertheless, in the logic IS4, $\Diamond^\bullet$ is the first source of non-termination, and for this we employ the same method that is commonly used for classical S4. However, whereas in the case of classical S4, the loop detection completes the proof search in the current branch, here we have to continue proof search. For this reason we realize the loop by creating a cluster.

In Chapter 5, we proved that the rewrite relation $\rightrightarrows_{\Diamond}$ terminates for some logics that do not include transitivity. In order to ensure termination for IS4 we need to add to Definition 5.2.3 the mentioned loop detection.

Definition 6.2.7 (Having no past)**.** A label x in a sequent $\mathfrak{G}$ ***has no past*** if and only if $y{\leq_{\mathfrak{G}}}x$ implies $y = x$ for all y.

Definition 6.2.8 (4-Saturation)**.** We define a binary relation $\overset{4}{\rightsquigarrow}_{\Diamond}$ on sequents as follows: $\mathfrak{G} \overset{4}{\rightsquigarrow}_{\Diamond} \mathfrak{G}'$ if and only if there is a label y and a formula A, such that $\mathfrak{G}$,$y{:}\Diamond A^\bullet$ is IS4-unhappy, and all labels $u \neq y$ such that $uR_{\mathfrak{G}}y$ but not $yR_{\mathfrak{G}}u$ are IS4-almost happy, and

1) Option 1: there is a label $x \neq y$ such that $x \sim y$, but $\mathfrak{G}$,$x{:}\Diamond A^\bullet$ is IS4-happy, $xR_{\mathfrak{G}}y$, and every label u with $xR_{\mathfrak{G}}u$ has no pasts in $\mathfrak{G}$. Then $\mathfrak{G}'$ is obtained from $\mathfrak{G}$ by first substituting x for each occurrence of y and then closing the resulting $R_{\mathfrak{G}}$ under transitivity. Or

2) Option 2: There is no such label x. Then $\mathfrak{G}'$ is obtained from $\mathfrak{G}$ by first adding $z{\leq}z$, yRz, $z{:}A^\bullet$ for some fresh label z, and then closing the resulting $R_{\mathfrak{G}}$ under transitivity and reflexivity.

For sets $\mathfrak{G}$ and $\mathfrak{G}'$ of sequents, we write $\mathfrak{G} \overset{4}{\rightrightarrows}_{\Diamond} \mathfrak{G}'$ iff $\mathfrak{G}$ is $\mathsf{IS4}$-semi-saturated, $\mathfrak{G} \overset{4}{\rightsquigarrow}_{\Diamond} \mathfrak{G}'$ for some $\mathfrak{G} \in \mathfrak{G}$ and $\mathfrak{G}'$ is a semi-saturation of $\Big(\mathfrak{G} \setminus \{\mathfrak{G}\}\Big) \cup \{\mathfrak{G}'\}$.[1] We write $\overset{4}{\rightrightarrows}^{*}_{\Diamond}$ for the transitive and reflexive closure of $\overset{4}{\rightrightarrows}_{\Diamond}$. If $\mathfrak{G} \overset{4}{\rightrightarrows}^{*}_{\Diamond} \mathfrak{G}'$ and $\mathfrak{G}'$ is in normal form with respect to $\overset{4}{\rightrightarrows}_{\Diamond}$, then $\mathfrak{G}'$ is called a $\mathsf{4}$-***saturation*** of $\mathfrak{G}$.

Lemma 6.2.9. *The following statements hold:*

a) If a sequent $\mathfrak{G}$ is $\mathsf{IS4}$-structurally saturated, tree-layered and tree-clustered and $\mathfrak{G} \overset{4}{\rightsquigarrow}_{\Diamond} \mathfrak{G}'$, then the sequent $\mathfrak{G}'$ is $\mathsf{IS4}$-structurally saturated, tree-layered and tree-clustered.

b) If a sequent $\mathfrak{G}$ is $\mathsf{IS4}$-semi-saturated and $\mathfrak{G} \overset{4}{\rightsquigarrow}_{\Diamond} \mathfrak{G}'$ according to Option 1, then the sequent $\mathfrak{G}'$ is $\mathsf{IS4}$-semi-saturated.

Proof. To prove a), assume that a sequent $\mathfrak{G}$ is $\mathsf{IS4}$-structurally saturated, tree-layered, and tree-clustered. Let $\mathfrak{G} \overset{4}{\rightsquigarrow}_{\Diamond} \mathfrak{G}'$. From Definition 6.2.8, in Option 1, there are no new labels and y has no past in $\mathfrak{G}$, hence, the only change in $\leq$-relational atoms is that $y{\leq}y$ turns into $x{\leq}x$; for Option 2, the only new label is z with $z{\leq}z$ being the only new $\leq$-relational atom. Either way, the $(\leq\mathsf{rf})$- and $(\leq\mathsf{tr})$-structural saturation is ensured and the set of irreflexive $\leq$-relational atoms remains unchanged. The latter implies that the $(\mathsf{mon}^{\bullet})$-, (F_1)-, and (F_2)-structural saturation of $\mathfrak{G}'$ follows from the same properties of $\mathfrak{G}$. The $(R\mathsf{tr})$-structural saturation is explicitly enforced. So is $(R\mathsf{rf})$-structural saturation for Option 2, where it is necessary for the added label z. This completes the proof that $\mathfrak{G}'$ is $\mathsf{IS4}$-structurally saturated.

As the rewrite does not affect the layer structure of the sequent, $\mathfrak{G}'$ being tree-layered immediately follows from the same property of $\mathfrak{G}$.

To show that $\mathfrak{G}'$ is tree-clustered we consider the two options separately. In Option 1 of Definition 6.2.8, the rewrite performs a substitution of label x for label y and applies the transitive closure. By considering the shortest chains justifying new $R_{\mathfrak{G}'}$-links, and using the structural saturation of both $\mathfrak{G}$ and $\mathfrak{G}'$ along with $xR_{\mathfrak{G}}y$, it follows that:

$$uR_{\mathfrak{G}'}x \implies uR_{\mathfrak{G}}y; \tag{6.2}$$

$$xR_{\mathfrak{G}'}v \implies xR_{\mathfrak{G}}v; \tag{6.3}$$

$$uR_{\mathfrak{G}'}v \text{ but not } uR_{\mathfrak{G}}v \implies uR_{\mathfrak{G}}y \& xR_{\mathfrak{G}}v. \tag{6.4}$$

Let C'_x be the cluster in $\mathfrak{G}'$ that contains x. For any cluster C in $\mathfrak{G}$, either $y \in C$ and $C \setminus \{y\} \subseteq C'_x$ or $y \notin C$ and either $C \subseteq C'_x$ or C remains a cluster in $\mathfrak{G}'$. The case of $y \in C$ is easy. Suppose $y \notin C$, which is not a cluster in $\mathfrak{G}'$. Since all $R_{\mathfrak{G}}$-links unrelated to y are preserved in $\mathfrak{G}'$, cluster C can only grow. So there must be some labels $u \in C$ and $v \notin C$ such that $uR_{\mathfrak{G}'}v$, $vR_{\mathfrak{G}'}u$, but either not $uR_{\mathfrak{G}}v$ or not $vR_{\mathfrak{G}}u$. By (6.4), in the former case, $uR_{\mathfrak{G}}y$ and $xR_{\mathfrak{G}}v$, hence, $uR_{\mathfrak{G}'}x$ and $xR_{\mathfrak{G}'}v$, while in the latter case, $vR_{\mathfrak{G}}y$ and $xR_{\mathfrak{G}}u$, hence, $vR_{\mathfrak{G}'}x$ and $xR_{\mathfrak{G}'}u$. Either way, by structural saturation of $\mathfrak{G}'$, $u \in C'_x$, and $C \subseteq C'_x$. To show that $\mathfrak{G}'$ is tree-clustered (in Option 1 of Def. 5.2.3), we consider three clusters C_1, C_2, and C' from $\mathfrak{G}'$ such that $C_1R_{\mathfrak{G}'}C'$ and $C_2R_{\mathfrak{G}'}C'$ and show that C_1 and C_2 are $R_{\mathfrak{G}'}$-comparable. If any two of them are equal, this is trivial, so assume they are pairwise

[1] Strictly speaking, forming the semi-saturation is only necessary in the second case, but to simplify the presentation, we do it in both cases.

distinct. If none of them is C'_x, then they are also clusters in $\mathfrak{G}$ and it is sufficient to show that there is some cluster C^* in $\mathfrak{G}$ such that $C_1 R_{\mathfrak{G}} C^*$ and $C_2 R_{\mathfrak{G}} C^*$. Then C_1 and C_2 are $R_{\mathfrak{G}}$-comparable because $\mathfrak{G}$ is tree-clustered and the connection remains in $\mathfrak{G}'$, meaning that they are also $R_{\mathfrak{G}'}$-comparable. If both $C_1 R_{\mathfrak{G}} C'$ and $C_2 R_{\mathfrak{G}} C'$, then $C^* = C'$. If neither $C_1 R_{\mathfrak{G}} C'$ nor $C_2 R_{\mathfrak{G}} C'$, then it follows from (6.4) that $C_1 R_{\mathfrak{G}} C_y$ and $C_2 R_{\mathfrak{G}} C_y$ where C_y is the cluster in $\mathfrak{G}$ that contains y. So $C^* = C_y$. Finally, if, w.l.o.g. $C_1 R_{\mathfrak{G}} C'$ but not $C_2 R_{\mathfrak{G}} C'$, then $C_2 R_{\mathfrak{G}} C_y$ and $C_x R_{\mathfrak{G}} C'$ by (6.4) where C_x is the cluster in $\mathfrak{G}$ that contains x. Since $\mathfrak{G}$ is tree-clustered, either $C_x R_{\mathfrak{G}} C_1$ or $C_1 R_{\mathfrak{G}} C_x$. In the latter case, $C_1 R_{\mathfrak{G}} C_y$, so $C^* = C_y$. In the former case, we derive the $R_{\mathfrak{G}'}$-connection $C_2 R_{\mathfrak{G}'} C_1$ directly from $C_2 R_{\mathfrak{G}} C_y$ and $C_x R_{\mathfrak{G}} C_1$. It remains to consider the case when exactly one of the three clusters is C'_x and the other two are clusters of $\mathfrak{G}$. If $C' = C'_x$, then $C_1 R_{\mathfrak{G}} C_y$ and $C_2 R_{\mathfrak{G}} C_y$ by (6.2), so the above argument with $C^* = C_y$ suffices. Finally, if w.l.o.g. $C_1 = C'_x$, then $C_x R_{\mathfrak{G}} C'$ by (6.3). If $C_2 R_{\mathfrak{G}} C'$, then C_x is $R_{\mathfrak{G}}$-comparable to C_2 because $\mathfrak{G}$ is tree-clustered, and C'_x is $R_{\mathfrak{G}'}$-comparable to C_2. If $C_2 R_{\mathfrak{G}'} C'$ but not $C_2 R_{\mathfrak{G}} C'$, then $C_2 R_{\mathfrak{G}} C_y$ by (6.4), and $C_2 R_{\mathfrak{G}'} C'_x$. The situation in Option 2 of Definition 5.2.3 is much simpler: there a single new cluster $\{z\}$ is added to the clusters of $\mathfrak{G}$ and $C R_{\mathfrak{G}'} \{z\}$ iff $C = \{z\}$ or $C R_{\mathfrak{G}} C_y$. The only non-trivial new case to consider is when $C' = \{z\}$, $C_1 \neq \{z\}$, and $C_2 \neq \{z\}$, in which case $C_1 R_{\mathfrak{G}} C_y$ and $C_2 R_{\mathfrak{G}} C_y$, and C_1 and C_2 are $R_{\mathfrak{G}'}$-comparable because they are $R_{\mathfrak{G}}$-comparable. This completes the proof that $\mathfrak{G}'$ is tree-clustered.

To prove b), since most of the conditions were just shown in a), it is sufficient to additionally show that all formulas in $\mathfrak{G}'$ are $\mathsf{IS4}$-naively happy. Since Option 1 adds neither new formulas nor new labels, for most types of formulas their (naive) happiness is inherited from $\mathfrak{G}$. The only exceptions are $\Box^\bullet$- and $\Diamond^\circ$-formulas. The argument is the same for both. We show it for $\Box^\bullet$-formulas. Suppose $u R_{\mathfrak{G}'} v$ and $\mathfrak{G}', u{:}\Box C^\bullet$. Since no formulas are added in Option 1, also $\mathfrak{G}, u{:}\Box C^\bullet$. There are two possibilities: either $u R_{\mathfrak{G}} v$ already in $\mathfrak{G}$ or it was added to ensure the structural saturation of $\mathfrak{G}'$. In the former case, $\mathfrak{G}', v{:}C^\bullet$ and $\mathfrak{G}', v{:}\Box C^\bullet$ by the naive happiness of this layer in $\mathfrak{G}$. In the latter case, we have $u R_{\mathfrak{G}'} v$ but not $u R_{\mathfrak{G}} v$, hence, by (6.4), both $u R_{\mathfrak{G}} y$ and $x R_{\mathfrak{G}} v$. Therefore, $\mathfrak{G}, y{:}\Box C^\bullet$ naive happiness of the layer in $\mathfrak{G}$, so $\mathfrak{G}, x{:}\Box C^\bullet$ because $x \sim y$ in $\mathfrak{G}$, and $\mathfrak{G}, v{:}C^\bullet$ and $\mathfrak{G}, v{:}\Box C^\bullet$, again due to naive happiness. Both $C^\bullet$ and $\Box C^\bullet$ remain at v in $\mathfrak{G}'$. □

Lemma 6.2.10. *Let $\mathfrak{G}$ be a $\mathsf{IS4}$-semi-saturated sequent, let $\mathfrak{G} \overset{4}{\rightsquigarrow}_{\Diamond} \mathfrak{G}'$, and let $\boldsymbol{\mathfrak{G}}^*$ be a semi-saturation of $\{\mathfrak{G}'\}$. Then all $\mathsf{IS4}$-happy $\Diamond^\bullet$-formulas are preserved by the transition from $\mathfrak{G}$ to $\boldsymbol{\mathfrak{G}}^*$, i.e., for all sequents $\mathfrak{G}^* \in \boldsymbol{\mathfrak{G}}^*$ and for all labels u occurring in $\mathfrak{G}^*$, if $\mathfrak{G}, u{:}\Diamond C^\bullet$ is $\mathsf{IS4}$-happy, then so is $\mathfrak{G}^*, u{:}\Diamond C^\bullet$.*

Proof. Note that semi-saturation does not remove formulas or labels, nor does it add R-relational atoms. Hence, semi-saturation in Option 2 cannot spoil happiness of any $\Diamond^\bullet$-formula, and it is sufficient to show that no $\mathsf{IS4}$-happy $\Diamond^\bullet$-formula of $\mathfrak{G}$ becomes $\mathsf{IS4}$-unhappy in $\mathfrak{G}'$. For Option 2, this is trivial as, again, nothing is removed. For Option 1, the only problems relate to the removal of y and could potentially occur if $\mathfrak{G}, u{:}\Diamond C^\bullet$ was $\mathsf{IS4}$-happy because of $u R_{\mathfrak{G}} y$ and $\mathfrak{G}, y{:}C^\bullet$. However, in this case, $u R_{\mathfrak{G}'} x$ and $\mathfrak{G}', x{:}C^\bullet$ because $x \sim y$ in $\mathfrak{G}$, thus, happiness persists. □

Lemma 6.2.11. *All the following statements hold:*

a) If a set $\boldsymbol{\mathfrak{G}}$ of sequents is $\mathsf{IS4}$-semi-saturated and $\boldsymbol{\mathfrak{G}} \overset{4}{\rightrightarrows}_{\Diamond} \boldsymbol{\mathfrak{G}}'$, then $\boldsymbol{\mathfrak{G}}'$ is $\mathsf{IS4}$-semi-saturated.

b) A set $\mathfrak{G}$ of sequents is IS4*-saturated iff it is* IS4*-semi-saturated and in normal form w.r.t.* $\overset{4}{\rightsquigarrow}_{\Diamond}$.

Proof. To prove a) assume that the set $\mathfrak{G}$ is finite and all sequents in $\mathfrak{G}$ are IS4-semi-saturated. Observe that $\mathfrak{G}' = \big(\mathfrak{G} \setminus \{\mathfrak{G}\}\big) \cup \mathfrak{G}^*$ for some $\mathfrak{G} \overset{4}{\rightsquigarrow}_{\Diamond} \mathfrak{G}'$ where $\mathfrak{G}^*$ is as in Lemma 6.2.10. In Option 1 of Definition 6.2.8, we replace one semi-saturated sequent $\mathfrak{G}$ with another one sequent $\mathfrak{G}'$, which is IS4-semi-saturated by Lemma 6.2.9.b). Hence, the resulting set $\mathfrak{G}'$ is IS4-semi-saturated. On the other hand, observe that by definition of $\overset{4}{\rightrightarrows}_{\Diamond}$, after each transition $\overset{4}{\rightsquigarrow}_{\Diamond}$, we calculate semi-saturation for both options of Definition 6.2.8. Then $\mathfrak{G}'$ is IS4-semi-saturated.

To prove b), observe that $\mathfrak{G}$ being IS4-saturated by definition means that it is IS4-semi-saturated and all its formulas are IS4-almost happy. A layer is IS4-almost happy if and only if it is IS4-naively happy and all $\Diamond A^{\bullet}$-formulas occurring in it are IS4-happy. Saturation $\overset{4}{\rightrightarrows}_{\Diamond}$ can be applied to a IS4-semi-saturated set iff it has an IS4-unhappy $\Diamond A^{\bullet}$-formula. Hence, $\mathfrak{G}$ is IS4-saturated iff it is IS4-semi-saturated and in normal form with respect to $\overset{4}{\rightrightarrows}_{\Diamond}$. □

We can now prove that indeed the rewrite relation $\overset{4}{\rightrightarrows}_{\Diamond}$ is terminating for the logic IS4:

Lemma 6.2.12. *The rewrite relation $\overset{4}{\rightrightarrows}_{\Diamond}$ is terminating.*

Proof. Let us consider a sequent $\mathfrak{G} \in \mathfrak{G}$ and an arbitrary layer L of $\mathfrak{G}$, having a $R_{\mathfrak{G}}$-path π with some unhappy $\Diamond^{\bullet}$-formulas. Let us define a sequence $(\pi_i)_{i<\omega}$ where $\pi_0 = L$ and π_{i+1} is generated from π_i by one step of the rewrite $\overset{4}{\rightrightarrows}_{\Diamond}$ applied to some label in π_i. Assume that, starting from some label w in π_n, for some $n > 0$, all labels in $\bigcup_{j=n}^{\omega} \pi_j$ are generated by Option 2 of the rewrite (Definition 6.2.8). It holds that for each j, the set of labels in π_j is included in the set of labels in π_{j+1}. Moreover, for all i, all labels in π_i are IS4-naively happy. We shall prove that, after a finite number of steps, we reach a set π_k, for $j \leq k$, whose labels are all IS4-almost happy. Since there are only finitely many labels and finitely many $\Diamond^{\bullet}$-formulas in π_0, the labels which can be introduced in $\bigcup_{j=n}^{\omega} \pi_j$ belong to finitely many $\sim$-equivalence classes of labels.

As soon as a cluster in π_j contains at least one label for each $\sim$-equivalence class, then all the $\Diamond^{\bullet}$-formulas occurring in it are IS4-happy. Thus, it is enough to show we find such a cluster in a finite number of steps. Whenever Option 1 of the rewrite is applied to some label in a cluster C_j in π_j, the corresponding cluster C_{j+1} in π_{j+1} might either have one element less than C_j (in case x and y belong to the same cluster) or be such that $C_j \subsetneq C_{j+1}$ (otherwise). However, the first case can occur only finitely many times, and cannot occur if all labels in C_j belong to pairwise disjoint $\sim$-equivalence classes. Moreover, it holds that whenever $C_j \subsetneq C_{j+1}$, C_{j+1} contains a label which does not belong to any $\sim$-equivalence class of labels in C_j. This is because whenever a label in a cluster contains an IS4-unhappy $\Diamond^{\bullet}$-formula, then the label k that would make that formula IS4-happy does not belong to the cluster, and needs to be created by Option 2 of the rewrite. Once created, children of k will be identified by Option 1 of the rewrite, and thus k will be added to the cluster. Thus, we generate clusters including an increasing number of labels belonging to different $\sim$-equivalence classes, until we obtain a cluster where all are represented, and whose labels are all IS4-almost happy. Since there are only finitely many paths in L, and since only finitely many new labels can be introduced at each label by the rewrite, we conclude that the rewrite relation $\overset{4}{\rightrightarrows}_{\Diamond}$ terminates. □

6.2.3 Lifting saturation

After having discussed how to saturate sets of sequents, we are now going to show how to deal with IS4-unhappy $A \supset B^\circ$ and $\Box A^\circ$. For this, we have to create new layers. The naive way would be to just copy the R-structure of the layer that contains the IS4-unhappy $A \supset B^\circ$ or $\Box A^\circ$ as it was done in the previous chapter. However, due to the presence of clusters, we sometimes need to create two copies of the cluster that contains the IS4-unhappy formula.

Construction 6.2.13 (4 Layer Lifting I)**.** Let $\mathfrak{G}$ be a IS4-saturated sequent, L be a layer in $\mathfrak{G}$, and let x be a label in L. Then x belongs to a cluster $C_x = \{x_1, \ldots, x_h\}$ with $h \geq 1$ and $x = x_m$ for some $1 \leq m \leq h$, which we often abbreviate as $m = 1..h$. In particular, $x_i R_{\mathfrak{G}} x_{i'}$ for any $i, i' = 1..h$. Let $\{y_1, \ldots, y_l\} = L \setminus C_x$ where $l \geq 0$. Then, if $h = 1$, let $\hat{L}$ be the set of fresh labels $\{\hat{y}_1, \ldots, \hat{y}_l, \hat{x}\}$. Otherwise, if $h > 1$, let $\hat{L} := \{\hat{y}_1, \ldots, \hat{y}_l, \hat{x}, \hat{x}'_1, \ldots, \hat{x}'_h, \hat{x}''_1, \ldots, \hat{x}''_h\}$ where again all labels are fresh. We define $\mathfrak{G}{\uparrow}^x_4$ to consist of:

1. relational atoms $v{\leq}v$ and vRv for all $v \in \hat{L}$;
2. for every $i = 1..l$, every $j = 1..h$, and every label w occurring in $\mathfrak{G}$:
   (a) relational atom $w{\leq}\hat{y}_i$ whenever $w{\leq_{\mathfrak{G}}}y_i$,
   (b) relational atom $w{\leq}\hat{x}$ whenever $w{\leq_{\mathfrak{G}}}x$,
   (c) only for $h > 1$: relational atoms $w{\leq}\hat{x}'_j$ and $w{\leq}\hat{x}''_j$ whenever $w{\leq_{\mathfrak{G}}}x_j$;
3. for all $i, i' = 1..l$ and $j, j' = 1..h$,
   (a) relational atom $\hat{y}_i R \hat{y}_{i'}$ whenever $y_i R_{\mathfrak{G}} y_{i'}$,
   (b) only for $h > 1$: relational atoms $\hat{y}_i R \hat{x}'_j$ and $\hat{y}_i R \hat{x}''_j$ whenever $y_i R_{\mathfrak{G}} x_j$,
   (c) relational atom $\hat{y}_i R \hat{x}$ whenever $y_i R_{\mathfrak{G}} x$,
   (d) only for $h > 1$: relational atoms $\hat{x}'_j R \hat{y}_i$ and $\hat{x}''_j R \hat{y}_i$ whenever $x_j R_{\mathfrak{G}} y_i$,
   (e) relational atom $\hat{x} R \hat{y}_i$ whenever $x R_{\mathfrak{G}} y_i$,
   (f) only for $h > 1$: relational atoms $\hat{x}'_j R \hat{x}'_{j'}$, $\hat{x}'_j R \hat{x}$, $\hat{x} R \hat{x}''_j$, and $\hat{x}''_j R \hat{x}''_{j'}$.
4. For every $i = 1..l$ and every $j = 1..h$, and for every formula C, add:
   (a) labelled formulas $\hat{y}_i{:}C^\bullet$ whenever $\mathfrak{G}, y_i{:}C^\bullet$,
   (b) only for $h > 1$: labelled formulas $\hat{x}'_j{:}C^\bullet$ and $\hat{x}''_j{:}C^\bullet$ whenever $\mathfrak{G}, x_j{:}C^\bullet$,
   (c) labelled formulas $\hat{x}{:}C^\bullet$ whenever $\mathfrak{G}, x{:}C^\bullet$.

Remark 6.2.14. Observe that Construction 6.2.13 is defined as Construction 5.2.8 plus the conditions required due to the presence of clusters.

This construction lifts a layer L of a sequent $\mathfrak{G}$ with respect to a label $x \in L$. If x is a singleton cluster, this is a simple lifting as the one implemented in Construction 5.2.8. Otherwise, if x is a non-singleton cluster, this cluster is duplicated and the lifting of x is put in between the two copies, as indicated in Figure 6.2 (ignore the label z for the moment).

In Construction 6.2.13, Points 1)–2) ensure that in sequent $\mathfrak{G} + \mathfrak{G}{\uparrow}^x_4$, the new layer is indeed above L as intended and ($\leq$tr), ($\leq$rf), and (Rrf) are satisfied; Point 3) ensures (Rtr),

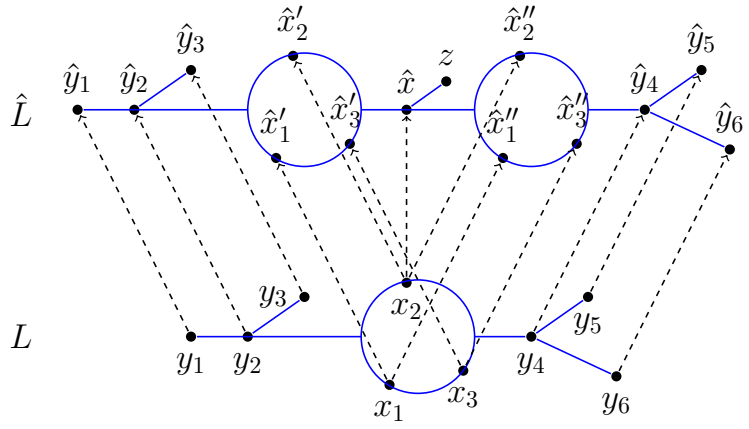


Figure 6.2: 4 Layer lifting as defined in Constructions 6.2.13 and 6.2.15

(F_1), and (F_2); Point 4) ensures ($\mathsf{mon}^\bullet$). Thus, $\mathfrak{G} + \mathfrak{G}{\uparrow}_4^x$ is tree-clustered, tree-layered, and $\mathsf{IS4}$-structurally saturated. It must be $\mathsf{IS4}$-almost happy due to the saturation of $\mathfrak{G}$ but may contain $\mathsf{IS4}$-unhappy $A \supset B^\circ$ and $\Box B^\circ$ formulas. To deal with them we need to adjust Construction 5.2.9 using the new layer lifting of Construction 6.2.13 which is introducing clusters. For a better understanding we redefine Construction 5.2.9 as follows:

Construction 6.2.15 (4 Layer Lifting II)**.** Let $\mathfrak{G}$ be a $\mathsf{IS4}$-saturated sequent with $\mathfrak{G}$,$x{:}F^\circ$ being $\mathsf{IS4}$-unhappy for some label x and some formula F of shape $A \supset B$ or $\Box B$. Let L be the layer of x. We define $\mathfrak{G}{\uparrow}_4^{x:F}$ as follows:

- If $F = A \supset B$, then we define $\mathfrak{G}{\uparrow}_4^{x:A\supset B}$ to be the sequent $\mathfrak{G}{\uparrow}_4^x$ to which we add labelled formulas $\hat{x}{:}A^\bullet$ and $\hat{x}{:}B^\circ$ and call $\hat{x}$ a ***suricata label of*** x (where $\hat{x}$ is as Construction 6.2.13);
- If $A = \Box B$, then we define $\mathfrak{G}{\uparrow}_4^{x:\Box B}$ to be the sequent $\mathfrak{G}{\uparrow}_4^x$ to which we add formulas zRz, $z{\leq}z$, and $z{:}B^\circ$ for a fresh label z, and additionally add relational atoms vRz whenever $v \in \hat{L}$ and $vR\hat{x}$ is in $\mathfrak{G}{\uparrow}_4^x$. Here z is called a ***suricata label of*** x. (where $\hat{x}, \hat{L}$ are as in Construction 6.2.13).

Remark 6.2.16. Observe that Construction 6.2.15 is the same as Construction 5.2.9 with the differences that we are working on $\mathsf{IS4}$-saturated sequent which are not only $\mathsf{IS4}$-structurally saturated and tree-layered but also they are tree-clustered; and we replace $\mathfrak{G}{\uparrow}^x$ (Construction 5.2.8) by $\mathfrak{G}{\uparrow}_4^x$ (Construction 6.2.15) which is including clusters.

As in Construction 5.2.9, the suricata label contains the white formula responsible for the happiness of the formulas $x{:}A \supset B^\circ$ and $x{:}\Box A^\circ$, and the result is still $\mathsf{IS4}$-structurally saturated and tree-layered and tree-clustered. Observe that Figure 6.2 shows the case of $x{:}\Box B^\circ$ with the additional fresh label z (that has no past).

6.2.4 Loop-saturation

We have now presented all the steps that make the sequent larger by adding labelled formulas or relational atoms. As a result, $\mathsf{IS4}$-unhappy formulas become $\mathsf{IS4}$-happy, or the sequent becomes $\mathsf{IS4}$-structurally saturated.

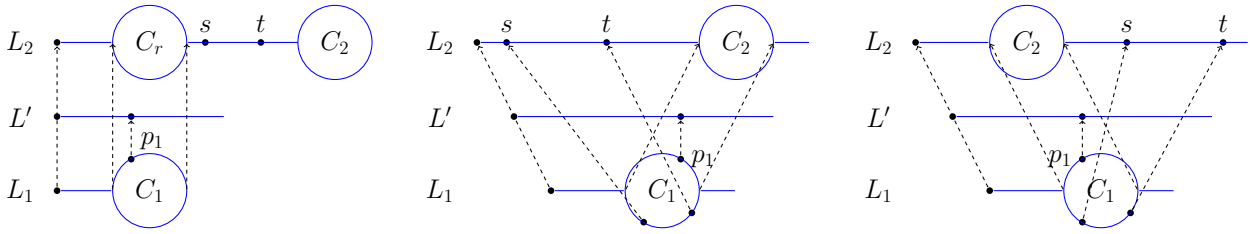


Figure 6.3: Left: Structure of an unhappy R-triangle loop defined in Definition 6.2.17. Middle and Right: Structure of an unhappy U-triangle loop defined in Definition 6.2.18

However, observe that, in general, new layers are larger than previous ones and there are two sources for the growth. First, in the case of an IS4-unhappy $x{:}\Box B^{\circ}$, a fresh label z is added, and second, if the label x is in a non-singleton cluster, then this cluster is duplicated. Both effects can be seen in Figure 6.2.

For this reason, we need to find a way to *shrink* a layer. This will be done by creating clusters similar to how Definition 6.2.8 does for $\Diamond^{\bullet}$. The difference is that this time the potential repetition will involve several layers rather than being local to a single layer. The difficulty is that some part of a layer L_1 will be repeated in a layer L_2 occurring above it, but in order to keep the sequent tree-layered, we cannot create clusters across several layers. The solution we implement here is to create a cluster inside layer L_2 for a part that would be repeated in a future layer, provided that we can repeat in layer L_2 what happened in layer L_1. We call such loops *triangle loops*. There are two kinds of such loops which distinguish the two ways a layer can grow discussed above. The first kind occurs if the cluster to be created in L_2 is in a part of L_2 that has no past in L_1. This could be caused for example by repetitions of $\Box^{\circ}$. We call these loops *R-triangle loops*. The second kind occurs when the repetition is caused by a repeated duplication of clusters in the layer lifting (see Figure 6.2). We call these loops *U-triangle loops*.

Before we give the formal definitions, observe that all new layers that are created in our algorithm for IS4 are of the shape $\mathfrak{G}{\uparrow}_4^{x:F}$, as defined in Construction 6.2.15, and each such layer contains exactly one suricata label. This can therefore be called *the suricata label of the layer*, and it is (immediately after the lifting) the only label that contains a white formula.[2]

Definition 6.2.17 (R-Triangle Loop)**.** Let $\mathfrak{G}$ be a IS4-saturated sequent with two layers L_1, L_2 such that $L_1 < L_2$. We say that clusters $C_1 \subseteq L_1$ and $C_2 \subseteq L_2$ form an ***R-triangle loop*** if and only if the following conditions hold:

1. $C_1 \sim C_2$;
2. there is a label $p_1 \in C_1$ such that there is a layer L' with $L_1 < L' \leq L_2$ that contains a suricata label of p_1;
3. there is a cluster C_r such that $C_1 {\leq_{\mathfrak{G}}} C_r$, and $C_r R_{\mathfrak{G}} C_2$, and no label $v \in L_2 \setminus C_r$ with $C_r \; R_{\mathfrak{G}} \; v \; R_{\mathfrak{G}} \; C_2$ has a past in L_1, i.e., $u {\not\leq_{\mathfrak{G}}} v$ for any $u \in L_1$.

The R-triangle loop is ***unhappy*** iff additionally:

[2]On the other hand, label x can have several suricata labels in other layers if there are more than one $\supset^{\circ}$ or $\Box^{\circ}$ formulas in x.

4. whenever $L_2 \leq L''$ for some layer L'', then $L_2 = L''$;

5. there are labels $s, t \in L_2 \setminus C_2$ such that $s \sim t$, and $s \neq t$, and $C_r \; R_{\mathfrak{G}} \; s \; R_{\mathfrak{G}} \; t \; R_{\mathfrak{G}} \; C_2$;

6. there is no suricata label u with $C_r \; R_{\mathfrak{G}} \; u \; R_{\mathfrak{G}} \; t$;

(Note that s and t may be in a common cluster.)

The Left of the Figure 6.3 illustrates this definition.

Definition 6.2.18 (U-Triangle Loop)**.** Let $\mathfrak{G}$ be a saturated sequent with two layers L_1, L_2 such that $L_1 < L_2$. We say that clusters $C_1 \subseteq L_1$ and $C_2 \subseteq L_2$ form a ***U-triangle loop*** iff the following conditions hold:

1. $C_1 \sim C_2$;

2. there is a label $p_1 \in C_1$ such that there is a layer L' with $L_1 < L' \leq L_2$ that contains a suricata label of p_1;

3. $C_1 \leq_{\mathfrak{G}} C_2$.

The U-triangle loop is ***unhappy*** iff additionally:

4. whenever $L_2 \leq L''$ for some layer L'', then $L_2 = L''$;

5. there are labels $s, t \in L_2 \setminus C_2$ such that $C_1 \leq_{\mathfrak{G}} s$, and $C_1 \leq_{\mathfrak{G}} t$, and $s \sim t$, and $s \neq t$, and either $C_2 \; R_{\mathfrak{G}} \; s \; R_{\mathfrak{G}} \; t$ or $s \; R_{\mathfrak{G}} \; t \; R_{\mathfrak{G}} \; C_2$;

6. there is no suricata label u with $s \; R_{\mathfrak{G}} \; u \; R_{\mathfrak{G}} \; t$;

(Again, s and t may be in the same cluster.)

The two possibilities envisioned by this definition, depending on whether $C_2 \; R_{\mathfrak{G}} \; s \; R_{\mathfrak{G}} \; t$ or $s \; R_{\mathfrak{G}} \; t \; R_{\mathfrak{G}} \; C_2$, are illustrated in Figure 6.3 (Middle and Right).

Informally, we speak of a *triangle loop* when we can reproduce the steps that started with the creation of L' and led to L_2. The loop is unhappy if we can observe some repetition in the new part of L_2 (with respect to L_1). If this is the case, we can collapse this repetition by creating a cluster, or shrinking an existing cluster, as follows:

Definition 6.2.19 (Loop-saturation)**.** Let $\mathfrak{G}$ and $\mathfrak{G}'$ be $\mathsf{IS4}$-saturated sequents. We write $\mathfrak{G} \rightsquigarrow_{\circlearrowleft} \mathfrak{G}'$ if and only if there is an unhappy R-triangle or U-triangle loop in $\mathfrak{G}$ where the labels s and t are as in Definition 6.2.17 or Definition 6.2.18 respectively and $\mathfrak{G}'$ is obtained from $\mathfrak{G}$ by substituting s for all occurrences of t and closing the resulting $R_{\mathfrak{G}}$ under transitivity. For sets $\mathfrak{S}$ and $\mathfrak{S}'$ of sequents, we write $\mathfrak{S} \rightsquigarrow\!\!\!\rightarrow_{\circlearrowleft} \mathfrak{S}'$ if and only if $\mathfrak{S}$ is $\mathsf{IS4}$-saturated, we have $\mathfrak{G} \rightsquigarrow_{\circlearrowleft} \mathfrak{G}'$ for some $\mathfrak{G} \in \mathfrak{S}$, and $\mathfrak{S}' = \big(\mathfrak{S} \setminus \{\mathfrak{G}\}\big) \cup \{\mathfrak{G}'\}$. We write $\rightsquigarrow\!\!\!\rightarrow^*_{\circlearrowleft}$ for the transitive and reflexive closure of $\rightsquigarrow\!\!\!\rightarrow_{\circlearrowleft}$. If $\mathfrak{S} \rightsquigarrow\!\!\!\rightarrow^*_{\circlearrowleft} \mathfrak{S}'$ and $\mathfrak{S}'$ is in normal form with respect to $\rightsquigarrow\!\!\!\rightarrow_{\circlearrowleft}$, then $\mathfrak{S}'$ is called a ***loop saturation*** of $\mathfrak{S}$.

As before, this term is justified because, a loop saturation of a $\mathsf{IS4}$-saturated set of sequents is saturated. We need to prove now that the rewrite relation $\rightsquigarrow\!\!\!\rightarrow_{\circlearrowleft}$ is terminating.

Lemma 6.2.20. *It holds that:*

a) If a set $\mathbf{G}$ *of sequents is* $\mathsf{IS4}$*-saturated and* $\mathbf{G} \rightsquigarrow_{\circlearrowleft} \mathbf{G}'$*, then* $\mathbf{G}'$ *is* $\mathsf{IS4}$*-saturated.*

b) The rewrite relation $\rightsquigarrow_{\circlearrowleft}$ *is terminating.*

Proof. To prove a) assume that $\mathbf{G}$ is saturated. Let us consider a sequent $\mathcal{G} \in \mathbf{G}$ such that $\mathcal{G} \rightsquigarrow_{\circlearrowleft} \mathcal{G}'$. Then, $\mathcal{G}$ contains either an unhappy R-triangle loop or an unhappy U-triangle loop. Sequent $\mathcal{G}'$ is obtained from $\mathcal{G}$ by substituting label s for all occurrences of label t and then closing R under transitivity (Definition 6.2.19). By assumption, $\mathcal{G}$ it is $\mathsf{IS4}$-structurally saturated, tree-layered and tree-clustered and all its formulas are $\mathsf{IS4}$-almost happy.

Tree-layered: Since the rewrite does not affect the layer structure of the sequent, we immediately conclude that $\mathbf{G}'$ is tree-layered.

Tree-clustered: The proof that $\mathcal{G}'$ is tree-clustered proceeds in the same way as the proof of case a) of Lemma 6.2.11, in the case of Option 1 of Definition 6.2.8. There, label x is substituted for label y and the proof relies on the fact that $x \sim y$ and $xR_{\mathcal{G}}y$. These same conditions hold for s and t, and therefore the same proof strategy applies.

$\mathsf{IS4}$*-structurally-saturated*: To prove that $\mathcal{G}'$ is $\mathsf{IS4}$-structurally saturated, observe that the rewrite does not introduce new labels, and the only changes in $\leq$-relational atoms are that $t{\leq}t$ turns into $s{\leq}s$ and, for any $k \neq t$, if $k{\leq_{\mathcal{G}}}t$, then $k{\leq_{\mathcal{G}'}}s$. The $(R\mathsf{tr})$-structural saturation of $\mathcal{G}'$ is explicitly enforced, while the $(\leq\mathsf{rf})$- and $(R\mathsf{rf})$-structural saturation are not affected by the removal of label t. The only non-trivial case for $(\leq\mathsf{tr})$-structural saturation is when for some label w, $w{\leq_{\mathcal{G}'}}k$ and $k{\leq_{\mathcal{G}'}}s$ because $k{\leq_{\mathcal{G}}}t$ where $k \neq s$. Then k and w belong to unaffected layers, hence, $w{\leq_{\mathcal{G}}}k$. Thus, $w{\leq_{\mathcal{G}}}t$ by $(\leq\mathsf{tr})$-structural saturation of $\mathcal{G}$, and $w{\leq_{\mathcal{G}'}}s$ by construction. The $(\mathsf{mon}^{\bullet})$-structural saturation follows from $s \sim t$. Finally, for (F_1)-, and (F_2)-structural saturation of $\mathcal{G}'$, the most non-trivial case is when, for one of the two, $kR_{\mathcal{G}'}w$ and $k{\leq_{\mathcal{G}'}}s$ because $k{\leq_{\mathcal{G}}}t$ where $k \neq s$. Again k and w belong to unaffected layers, hence, $kR_{\mathcal{G}}w$. By the same structural saturation property of $\mathcal{G}$, there exists a label z such that $w{\leq_{\mathcal{G}}}z$ and $tR_{\mathcal{G}}z$. If $t \neq z$, then $sR_{\mathcal{G}'}z$ and $w{\leq_{\mathcal{G}'}}z$, so the same label z fulfils the conditions. If $t = z$, then $w{\leq_{\mathcal{G}'}}s$, which together with $sR_{\mathcal{G}'}s$ means that s fulfills the conditions. The remaining case is symmetric. This concludes the proof that $\mathcal{G}'$ is $\mathsf{IS4}$-structurally saturated.

Formulas are $\mathsf{IS4}$*-almost happy*: By assumption, $\mathcal{G}$ is $\mathsf{IS4}$-saturated. This means that all its formulas are $\mathsf{IS4}$-almost happy, that is, all formulas that are not of the shape $A \supset B^{\circ}$ or $\Box A^{\circ}$ are $\mathsf{IS4}$-happy. It remains to show that all formulas in $\mathcal{G}'$ are $\mathsf{IS4}$-almost happy. But almost happiness of labelled propositional formulas of $\mathcal{G}'$ is local and, hence, is not affected by the new R- and $\leq$-links. Happiness of $\Box^{\bullet}$- and $\Diamond^{\circ}$-formulas is shown as in the proof of case a) of Lemma 6.2.11.

This concludes the proof that $\mathbf{G}'$ is $\mathsf{IS4}$-saturated.

To prove b), observe that, unlike in the case of $\overset{4}{\rightsquigarrow}_{\Diamond}$ (Definition 6.2.8), $\rightsquigarrow_{\circlearrowleft}$ never introduces new labels in the sequents on which it operates. Whenever $\rightsquigarrow_{\circlearrowleft}$ is applied to a sequent $\mathcal{G}$, the number of labels occurring in one of the layers of $\mathcal{G}$ is reduced by one because $s \neq t$. Therefore, for any layer L in $\mathcal{G}$ containing $n \geq 1$ labels, transformation $\rightsquigarrow_{\circlearrowleft}$ can be applied at most $n-1$ times to the labels occurring in L. Since saturated set $\mathbf{G}$ is finite, the rewrite $\rightsquigarrow_{\circlearrowleft}$ is terminating. □

Definition 6.2.21 (Full Saturation)**.** Let $\mathbf{G}$ be a set of sequents. Let $\mathbf{G}'$ be a semi-saturation of $\mathbf{G}$, and $\mathbf{G}''$ be a 4-saturation of $\mathbf{G}'$, and $\mathbf{G}'''$ be a loop-saturation of $\mathbf{G}''$. Then $\mathbf{G}'''$ is a ***full saturation*** of $\mathbf{G}$.

0. Given a formula F, define $\mathfrak{G}_0(F)$ to be the sequent $r{\leq}r \implies r{:}F$ and let $\mathfrak{G}'_0 := \{\mathfrak{G}_0(F)\}$.

1. For the set $\mathfrak{G}'_i$, calculate a full saturation $\mathfrak{G}_i$.

2. If all sequents in $\mathfrak{G}_i$ are axiomatic, then terminate.
   The formula F is provable and we can give a proof of $\implies r{:}F$ in $\mathsf{labX}'_{\leq}$.

3. Otherwise, pick a non-axiomatic sequent $\mathfrak{G}_i \in \mathfrak{G}_i$.

   (a) Pick an allowed formula $x{:}F^{\circ}$, compute $\mathfrak{G}_i{\uparrow}_4^{x:F^{\circ}}$ and set $\mathfrak{G}'_{i+1} = (\mathfrak{G}'_i \setminus \{\mathfrak{G}_i\}) \cup \{\mathfrak{G}_i + \mathfrak{G}_i{\uparrow}_4^{x:F^{\circ}}\}$ and go to Step 1.

   (b) If we do not have any more allowed formulas, then terminate.
   The formula F is not provable, and the sequent $\mathfrak{G}_i$ defines a countermodel (see Section 5.3).

Figure 6.4: Proof search algorithm for $\mathsf{IS4}$

Lemma 6.2.22. *If a set $\mathfrak{G}$ of sequents is $\mathsf{IS4}$-saturated and $\mathfrak{G}'''$ its full saturation, then $\mathfrak{G}'''$ is $\mathsf{IS4}$-saturated.*

Proof. By definition of full saturation, we have that $\mathfrak{G}'$ is a semi-saturation of $\mathfrak{G}$ and by Lemma 6.2.11, b), $\mathfrak{G}'$ is $\mathsf{IS4}$-saturated. We have that $\mathfrak{G}''$ is a 4-saturation of $\mathfrak{G}'$. $\mathfrak{G}'''$ is a loop-saturation of the $\mathsf{IS4}$-saturated set of sequents $\mathfrak{G}''$ and by Lemma 6.2.20, a), $\mathfrak{G}'''$ is $\mathsf{IS4}$-saturated. □

6.2.5 Proof search algorithm

Figure 6.4 shows the search algorithm for $\mathsf{IS4}$. Observe that there are only two different changes with respect to the algorithm in Figure 5.2: (i) in Step 1 we use the full saturation, since to ensure termination in $\mathsf{IS4}$ we need two add other loop detections (one for the $\Diamond^{\bullet}$-formulas and the loop saturation for every unhappy triangle loop); (ii) in Step 3.a) the construction $\mathfrak{G}_i{\uparrow}_4^{x:F^{\circ}}$ of the lifting of the formula F which now it involves clusters. The allowed formulas we are looking for in the search algorithm of Figure 6.4, are the same allowed formulas as in the algorithm in Figure 5.2 which are defined in Definition 5.2.13. This means we are using the same definitions for simulation of layers in this chapter (see Definitions 5.2.11 and 5.2.12).

Example 6.2.1 (Valid formula in $\mathsf{IS4}$)**.** Let us consider the next valid formula in $\mathsf{IS4}$: $\Box(\Diamond((c \supset \Diamond b) \supset \bot) \wedge \Diamond b) \supset \bot$, and let $A = (c \supset \Diamond b) \supset \bot$. Figure 6.5 represents one sequent generated by the algorithm. Let $\Gamma^{\bullet}$ be the set $\{\Box(\Diamond A \wedge \Diamond b)^{\bullet}, \Diamond A \wedge \Diamond b^{\bullet}, \Diamond A^{\bullet}, \Diamond b^{\bullet}\}$. To each label in the Figure (except 1) we associate $\Gamma^{\bullet}$, plus the formulas explicitly displayed next to the node in the Figure. The following $\leq$-relations are not displayed but are present in the sequent: $3{\leq}10$, $4{\leq}11$, $6{\leq}12$, $6{\leq}15$, $7{\leq}13$ and $7{\leq}16$.

At layer L_1 the search on both R-branches stops in virtue of Option 1 of Definition 5.2.3 (saturation): there, 5 is replaced with 3, and 8 is replaced with 6. This generates two non-

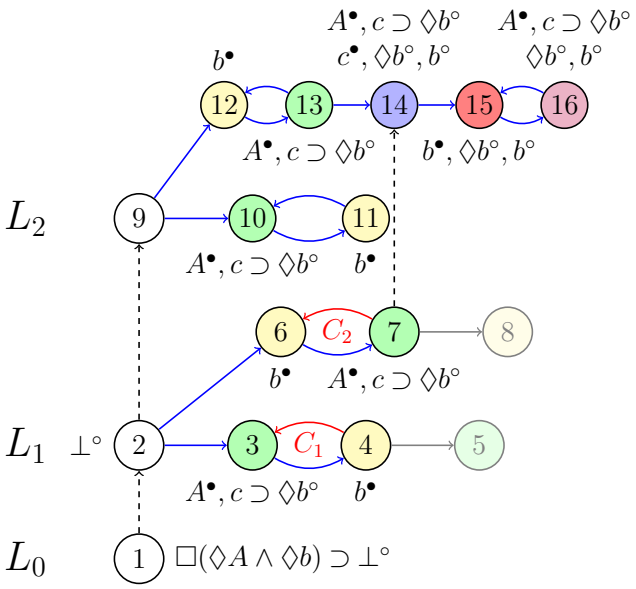


Figure 6.5: Formula $\Box(\Diamond A \wedge \Diamond b) \supset \bot$, with $A = (c \supset \Diamond b) \supset \bot$.

singleton clusters: $C_1 = \{3, 4\}$ and $C_2 = \{6, 7\}$. Then, since all labels in L_1 are $\mathsf{IS4}$-almost happy, lifting saturation is applied: the rewrite "lifts" one copy of the layer for each $\supset^\circ$-formula present in the sequent. We only represent one of such layers, L_2, generated from formula 7:$c \supset \Diamond b^\circ$. Since 7 belongs to the cluster C_2, the cluster is duplicated in L_2, following Constructions 6.2.13 and 6.2.15 and generating clusters $\{12, 13\}$ and $\{15, 16\}$. The label 14 is a suricata label.

Then, after a semi-saturation step, we have that both 15:$b^\bullet$ and 15:b°. Thus we stop the search along this branch. All the sequents manipulated by the algorithm eventually result in axiomatic sequents, and the algorithm stops in Step 2. Then, according to Theorem 6.4.19, the formula at the root is a theorem of $\mathsf{IS4}$. Indeed, its derivation in $\mathsf{labIS4}_{\leq}$ can be found in Figure 5.1 (Left).

6.3 Countermodel construction

The countermodel construction is exactly the same as the one presented in Chapter 5, Section 5.3. In other words, the proof of the following Theorem 6.3.1 is the same proof of Theorem 5.3.1:

Theorem 6.3.1. *If the algorithm shown in Figure 6.4 terminates in Step 3.b), then the formula F is not a theorem of* $\mathsf{IS4}$.

Subsequently, we provide examples of non-provable formulas in $\mathsf{IS4}$ where we construct a countermodel for them and the different triangle loops can be clearly observed.

Example 6.3.1 (Non-valid formula, R-triangle loop)**.** Figure 6.6 contains a (partial) diagrammatic representation of one sequent generated by the algorithm, when run on the valid $\mathsf{IS4}$ formula from Example 5.1, that is, $\Box(A \wedge B) \supset \bot$ with $A = \Box a \supset \bot$ and $B = (a \supset \bot) \supset \bot$.

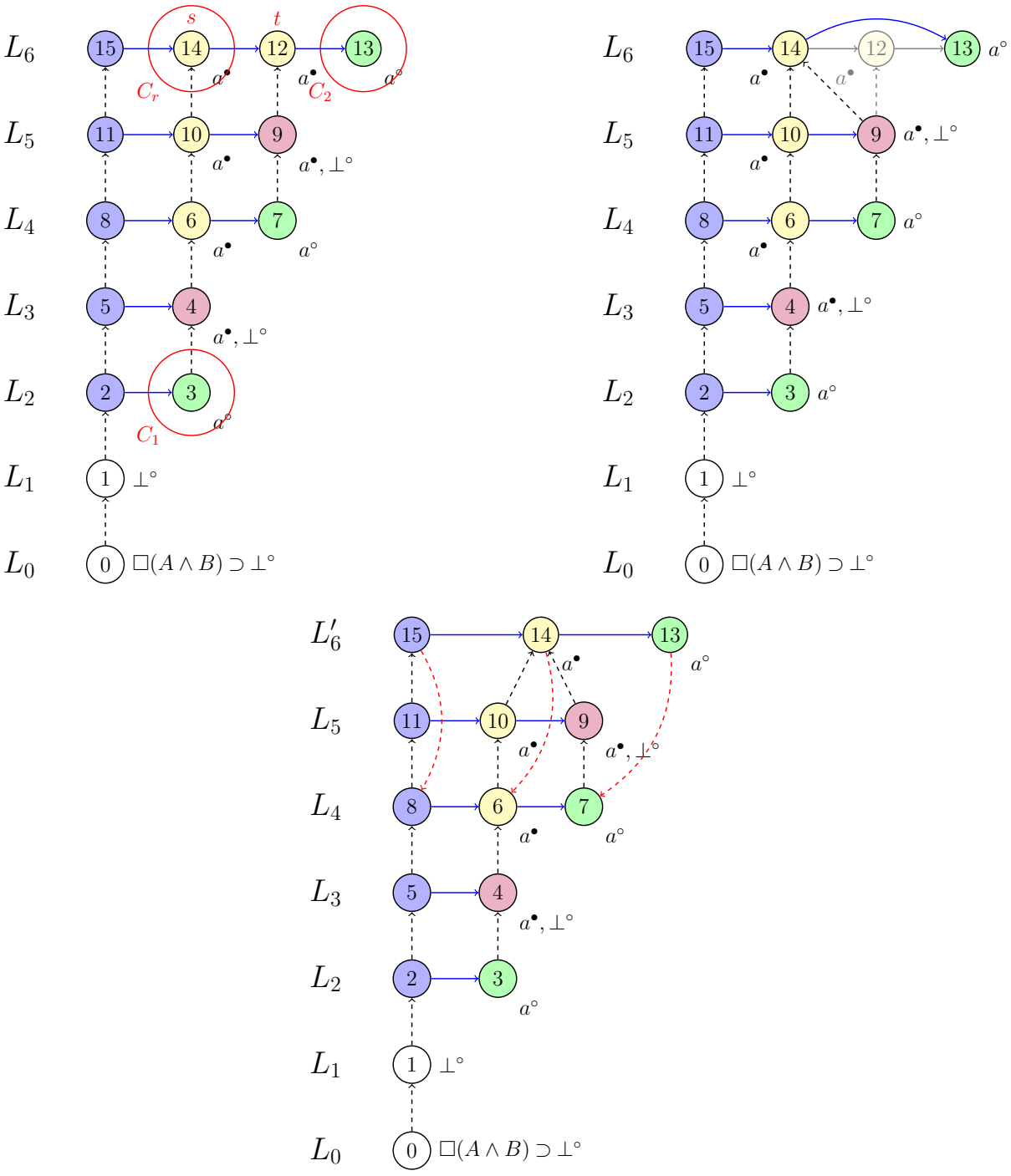


Figure 6.6: Formula $\Box(A \wedge B) \supset \bot$ with $A = \Box a \supset \bot$ and $B = (a \supset \bot) \supset \bot$. A loop saturation step is applied to the leftmost sequent, and the result of the saturation is the lowest sequent.

Let $\Gamma^\bullet = \{\Box(A \wedge B)^\bullet, A \wedge B^\bullet, A^\bullet, B^\bullet, \Box a^\circ, a \supset \bot^\circ\}$. To each label in the figure (except 0), we associate $\Gamma^\bullet$, plus the formulas explicitly displayed next to the node. To be precise, the figure displays only one $\leq$-branch of the sequent: formulas $\Box a^\circ$ and $a \supset \bot^\circ$ are in all

the labels, and every time they are IS4-unhappy they are also lifted, generating a new layer (which is not displayed). On the top left of Figure 6.6, at layer L_6 an unhappy R-triangle loop is detected: employing the terminology from Definition 6.2.17, we take $C_1 = \{3\}$, $C_2 = \{13\}$, $s = 14$, $C_r = \{s\}$ and $t = 12$.

Then, as it is represented on the top right of the Figure 6.6, the label 12 is replaced with label 14 creating the corresponding links: $14R13$ and $9 \leq 14$.

On the lowermost sequent of Figure 6.6, 12 disappeared due to the loop saturation and the cluster $\{14\}$ remains unchanged. Proof search along the $\leq$-branch depicted stops: this is because layer $L_{6'}$ can be simulated by layer L_4 (see Definitions 5.2.11 and 5.2.12). By adding relational atoms $15{\leq}8$, $14{\leq}6$ and $13{\leq}7$ to the sequent (Theorem 6.3.1), which are the dashed red arrows pointing downwards in the lowermost sequent of the Figure, we obtain (a part of) the countermodel for our formula. To complete the countermodel, we need to take into account the layers generated from IS4-unhappy $\Box^\circ$ and $\supset^\circ$ formulas present in the sequent. Specifically, $8{:}a \supset \bot^\circ$ is IS4-unhappy. This formula gives rise to more $\leq$-branches in the sequent (but all branches are finite, and there is only a finite number of them).

Example 6.3.2 (Non-valid formula, U-triangle loop)**.** Let us consider the following formula: $\Box(\Diamond((a \supset b) \supset \bot) \wedge \Diamond c) \supset \bot$, and let $D = (a \supset b) \supset \bot$. Figure 6.7 illustrates one sequent generated by the search algorithm. Let $\Gamma^\bullet = \{\Box(\Diamond D \wedge \Diamond c)^\bullet, \Diamond D \wedge \Diamond c^\bullet, \Diamond D^\bullet, \Diamond c^\bullet\}$. To each label we associate the following sets, which have the same colors in the Figure:

1	$\Gamma^\bullet \cup \{\Box(\Diamond D \wedge \Diamond c) \supset \bot^\circ\}$
2	$\Gamma^\bullet \cup \{\bot^\circ\}$
$10 \sim 19 \sim 29$	$\Gamma^\bullet$
$3 \sim 8 \sim 6 \sim$ $18 \sim 15 \sim 12 \sim$ $28 \sim 23 \sim 21$	$\Gamma^\bullet \cup \{D^\bullet, a \supset b^\circ\}$
$4 \sim 9 \sim 7 \sim$ $17 \sim 14 \sim 13 \sim$ $27 \sim 24 \sim 22 \sim$	$\Gamma^\bullet \cup \{c^\bullet\}$
$16 \sim 26 \sim 25$	$\Gamma^\bullet \cup \{D^\bullet, a^\bullet, a \supset b^\circ\}$
$5 \sim 11 \sim 20$	$\Gamma^\bullet \cup \{D^\bullet, a^\bullet, b^\circ, a \supset b^\circ\}$

As in the previous example, only one $\leq$-branch is represented. Moreover, we have not pictured a second R-branch originating from 2 by saturation, to make formula $2{:}\Diamond c^\bullet$ IS4-happy. The labels originated by lifting this branch are present in all layers.

As in Example 6.2.1, in layer L_1, after saturation, a non-singleton loop is created: $\{3, 4\}$. Then, layer L_2 is generated by lifting saturation, which duplicates the cluster. The process is repeated to generate L_3. At L_3, an unhappy U-triangle loop is found. Using the terminology from Definition 6.2.18, we take $C_1 = \{3, 4\}$, $C_2 = \{12, 13\}$, $s = 18$ and $t = 15$. Thus, we substitute 15 with 18. Moreover, a second unhappy U-triangle loop is found, by taking $s' = 17$ and $t' = 14$ (refer to the topmost sequent of Figure 6.7). Layer $L_{3'}$ is the result of the loop saturation: it contains a cluster $\{18, 17, 16\}$. The algorithm produces a sequent consisting of layers L_0 - L_2, $L_{3'}$, and continues (this refers to the middle sequent in Figure 6.7).

Then, after a lifting saturation, layer L_4 is generated. Here, three unhappy U-triangle loops are present: take $C_1 = \{3, 4\}$, $C_2 = \{21, 22\}$, and then $s = 28$ and $t = 23$, $s' = 27$

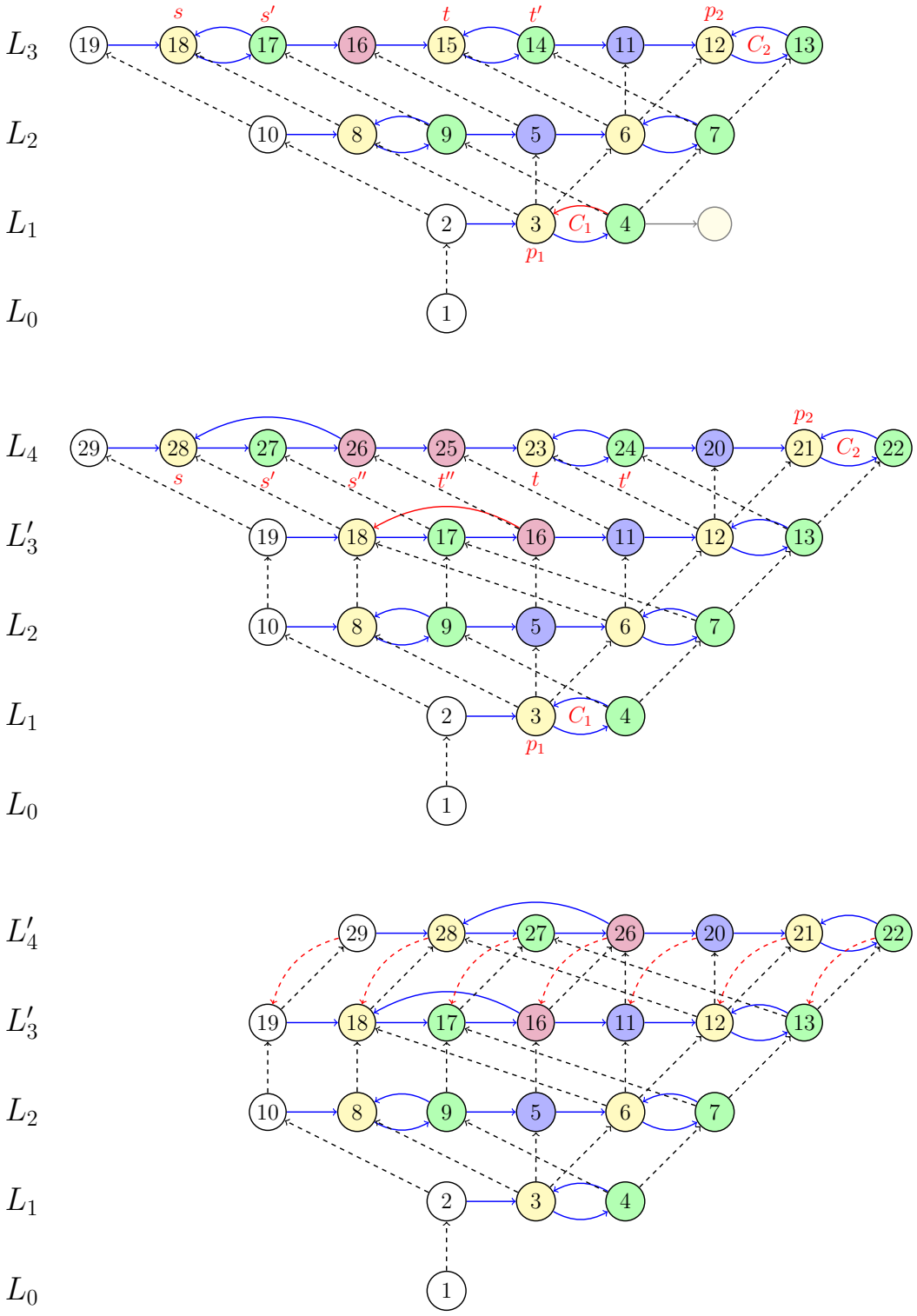


Figure 6.7: Formula $\Box(\Diamond D \wedge \Diamond c) \supset \bot$, with $D = (a \supset b) \supset \bot$. A loop saturation step is applied to the topmost sequent, and the result of the saturation is represented in the middle sequent. Then, a loop saturation is also applied to the middle sequent, and the result is the lowermost sequent.

$$4^\bullet\ \frac{\mathcal{R}, xRy, x{:}\Box A, y{:}\Box A, \Gamma \Longrightarrow \Delta}{\mathcal{R}, xRy, x{:}\Box A, \Gamma \Longrightarrow \Delta} \qquad 4^\circ\ \frac{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A, y{:}\Diamond A}{\mathcal{R}, xRy, \Gamma \Longrightarrow \Delta, x{:}\Diamond A}$$

Figure 6.8: Admissible rules

and $t' = 24$ and $s'' = 26$ and $t'' = 25$. After the loop saturation, we obtain layer $L_{4'}$, which can be simulated by layer $L_{3'}$ (this refers to the lowermost sequent in Figure 6.7). By adding relational atoms $29{\leq}19$, $28{\leq}18$, $27{\leq}17$, $26{\leq}16$, $20{\leq}11$, $21{\leq}12$ and $22{\leq}13$ we have a (partial) countermodel for the formula at the root. To complete the countermodel, we need to include the layers generated by lifting the layers of the sequent in correspondence to each remaining $\mathsf{IS4}$-unhappy formula $(a \supset b)^\circ$.

6.4 Proof construction: Unfolding

Let us now turn to the case when the algorithm terminates in Step 2. Then all sequents in $\mathfrak{S}_i$ are axiomatic, and we want to construct a proof of $\Longrightarrow r{:}F$ in $\mathsf{labIS4}_{\leq}$. For this we are going to simulate the steps of the algorithm by applying the inference rules of $\mathsf{labIS4}'_{\leq}$, which is defined as $\mathsf{labIS4}'_{\leq} = \mathsf{labIS4}_{\leq} \cup \{\mathsf{mon}^\bullet, \mathsf{weak}, 4^\bullet, 4^\circ, \mathsf{cont}^\bullet, \mathsf{cont}^\circ\}$, where the rules $\mathsf{mon}^\bullet, \mathsf{weak}, 4^\bullet, 4^\circ, \mathsf{cont}^\bullet, \mathsf{cont}^\circ$ are admissible (see Proposition 4.2.3, Lemma 4.5.2, Theorem 4.6.1 and Remark 4.2.4 in Chapter 4) and they are also presented in Figures 5.5 and 6.8:

Corollary 6.4.1. *The rules* $\mathsf{mon}^\bullet, \mathsf{weak}, 4^\bullet, 4^\circ, \mathsf{cont}^\bullet$ *and* cont° *are admissible for* $\mathsf{labIS4}_{\leq}$. *Therefore a formula* A *is derivable in* $\mathsf{labIS4}'_{\leq}$ *if and only if it is derivable in* $\mathsf{labIS4}_{\leq}$.

Our search algorithm was designed as organized proof search in $\mathsf{labIS4}'_{\leq}$, and most steps can indeed be executed by applying the rules of $\mathsf{labIS4}'_{\leq}$ as it was the case for the other logics in Chapter 5, Section 5.4. However, for $\mathsf{IS4}$ is different since we incorporated clusters. In particular, the difficulties come from the fact that the sequents produced by $\mathsf{labIS4}'_{\leq}$ do not have non-singleton clusters. We call those sequents *proper*.

Definition 6.4.2 (Vertical sequent)**.** A layered sequent $\mathfrak{G}$ is ***vertical*** if and only if for all labels u, u', v, and v' in $\mathfrak{G}$, (a) if $u{\leq_{\mathfrak{G}}}v$, and $u{\leq_{\mathfrak{G}}}v'$, and $vR^{\leftrightarrow}_{\mathfrak{G}}v'$, then $v = v'$, and (b) if $u{\leq_{\mathfrak{G}}}v$, and $u'{\leq}v$, and $uR^{\leftrightarrow}_{\mathfrak{G}}u'$, then $u = u'$, i.e., for each label there is at most one future per layer and at most one past per layer.

Definition 6.4.3 (Proper layer/sequent)**.** A layer L in a tree-clustered sequent $\mathfrak{G}$ is called ***proper*** if and only if all clusters $C \subseteq L$ are singletons. A tree-clustered sequent $\mathfrak{G}$ is ***proper*** if and only if it is tree-layered and vertical and all its layers are proper.

The basic idea of constructing our proof is to mimic the algorithm with a derivation that works "layer by layer". The key observation is that whenever we create a cluster in the algorithm, we can unfold it, this means that we repeat this cluster arbitrarily often in an actual proof with only proper sequents. And this property is preserved when we lift the cluster to the next layer.

Notation 6.4.4. We use $\ell(\mathfrak{G})$ to denote the set of labels occurring in a sequent $\mathfrak{G}$.

Definition 6.4.5 (Unfolding)**.** For $n \in \mathbb{N}$, a proper sequent $\hat{\mathfrak{G}}$ is called an ***n-unfolding*** of a sequent $\mathfrak{G}$ if and only if there is a binary ***n-unfolding relation*** $\mathsf{U} \subseteq \ell(\mathfrak{G}) \times \ell(\hat{\mathfrak{G}})$ that has the following properties, where we assume $x\mathsf{U}\hat{x}$ and $y\mathsf{U}\hat{y}$ and write L_u to denote the layer of u:

(U1) *formula invariance*: $x \sim \hat{x}$;

(U2) *R-invariance*: if x and y are not in the same cluster in $\mathfrak{G}$, then $xR_{\mathfrak{G}}y$ iff $\hat{x}R_{\hat{\mathfrak{G}}}\hat{y}$;

(U3) *layer invariance*:$L_x \leq_{\mathfrak{G}} L_y$ iff $L_{\hat{x}} \leq_{\hat{\mathfrak{G}}} L_{\hat{y}}$;

(U4) *"no-past" invariance*: x has no past in $\mathfrak{G}$ iff $\hat{x}$ has no past in $\hat{\mathfrak{G}}$;

(U5) *singleton-cluster invariance*: if $C = \{z\}$ is a singleton cluster in $\mathfrak{G}$, then there is a unique $\hat{z} \in \ell(\hat{\mathfrak{G}})$ with $z\mathsf{U}\hat{z}$;

(U6) *non-singleton-cluster unfolding*: if C is a cluster in $\mathfrak{G}$ with $|C| = k \geq 2$, then for all $i = 1..k$ and $j = 1..n$ there are $z_i \in \ell(\mathfrak{G})$ and $\hat{z}_{i,j} \in \ell(\hat{\mathfrak{G}})$ such that we have $z_i\mathsf{U}\hat{z}_{i,j}$ and $C = \{z_1, \ldots, z_k\}$ and additionally $\hat{z}_{1,1}R \cdots R\hat{z}_{k,1}R\hat{z}_{1,2}R \cdots R\hat{z}_{k,2}R \cdots R\hat{z}_{1,n}R \cdots R\hat{z}_{k,n}$ in $\hat{\mathfrak{G}}$;

(U7) *injectivity*: if $\hat{x} = \hat{y}$, then $x = y$.

Note that every $(n+1)$-unfolding is also an n-unfolding and that any proper sequent is an n-unfolding of itself for any n, via the diagonal relation $\{(x, x) \mid x \in \ell(\mathfrak{G})\}$. We will make use of these facts throughout this section without mentioning them explicitly.

Definition 6.4.6 (Unfoldable)**.** A derivation $\mathcal{D}_n$ is called an ***n-unfolding of a set of sequents*** $\mathfrak{S}$ iff each premise of $\mathcal{D}_n$ is an n-unfolding of some sequent from $\mathfrak{S}$. For a formula F, a set $\mathfrak{S}$ of sequents is ***F-unfoldable*** if for every $n \in \mathbb{N}$ there is a derivation $\mathcal{D}_n$ in the system $\mathsf{labIS4}'_{\leq}$ such that the conclusion of $\mathcal{D}_n$ is $\mathfrak{G}_0(F)$ (as defined in Step 0 of the algorithm) and each premise of $\mathcal{D}_n$ is an n-unfolding of some sequent from $\mathfrak{S}$.

Clearly, $\mathfrak{S}_0 = \{\mathfrak{G}_0(F)\}$ is F-unfoldable. It now has to be shown that all operations performed in the algorithm, in particular, the rewrite relations $\rightsquigarrow_{\mathsf{s}}, \overset{4}{\rightsquigarrow}_{\Diamond}$ and $\rightsquigarrow_{\circlearrowleft}$, preserve this property. For $\rightsquigarrow_{\mathsf{s}}$, this is straightforward, as we simply apply the inference rules of $\mathsf{labIS4}'_{\leq}$ to match the semi-saturation steps. For $\overset{4}{\rightsquigarrow}_{\Diamond}$ and $\rightsquigarrow_{\circlearrowleft}$, we need to unfold the newly created clusters. The basic idea is to repeat all proof steps that lead to the first occurrence of the cluster n-times. In particular for $\overset{4}{\rightsquigarrow}_{\Diamond}$, we repeat the proof search steps that led from the first occurrence of the $\Diamond A^{\bullet}$ to the second one that causes the use of Option 1 in Definition 6.2.8. Finally, for the loop-saturation, we can, similarly to $\overset{4}{\rightsquigarrow}_{\Diamond}$, repeat all proof steps that lead to the triangle loop. Additionally, we need to lift copies of the branch $s - t$ to the new layer.

Definition 6.4.7 (Soundness)**.** A rewrite relation $\rightsquigarrow$ on sets of sequents is ***sound*** if and only if for every formula F, whenever $\mathfrak{S} \rightsquigarrow \mathfrak{S}'$ and $\mathfrak{S}$ is F-unfoldable, then $\mathfrak{S}'$ is also F-unfoldable.

Lemma 6.4.8. *The rewrite relation $\rightsquigarrow_{\mathsf{s}}$ is sound.*

Proof. Let $\mathfrak{S} \rightrightarrows_{\mathsf{s}} \mathfrak{S}'$ for sets $\mathfrak{S}$ and $\mathfrak{S}'$ of sequents and let $\mathfrak{S}$ be F-unfoldable for some formula F, i.e., for every n, there is a derivation $\mathcal{D}_n$ of $\mathfrak{G}_0(F)$ where each premise of $\mathcal{D}_n$ is an n-unfolding of some sequent from $\mathfrak{S}$. We need to show that $\mathfrak{S}'$ is F-unfoldable, i.e., for every n there is a derivation $\mathcal{D}'_n$ of $\mathfrak{G}_0(F)$ where each premise of $\mathcal{D}'_n$ is an n-unfolding of some sequent from $\mathfrak{S}'$.

It is sufficient to consider each way that $\mathfrak{S}'$ can be obtained from $\mathfrak{S}$ by one step listed in Definition 5.2.1. Depending on the step, we start with $\mathcal{D}'_N$ for some $N \geq n$ so that we have extra copies of each non-singleton cluster of $\mathfrak{G}$ and can form n copies of the new clusters from $\mathfrak{G}'$ (or $\mathfrak{G}''$), with the remaining copies discarded from U. The only sequent from $\mathfrak{S}$ that is not present in $\mathfrak{S}'$ is $\mathfrak{G}$, which is replaced in $\mathfrak{S}'$ by sequent $\mathfrak{G}'$ or by two sequents $\mathfrak{G}'$ and $\mathfrak{G}''$, depending on the step. Hence, the only premises of $\mathcal{D}_N$ that require to be extended are those that were n-unfoldings of $\mathfrak{G}$. All the other premises are N-unfoldings and, hence, n-unfoldings of respective unchanged sequents from $\mathfrak{S}'$.

For each premise $\hat{\mathfrak{G}}$ of $\mathcal{D}_N$ that is an N-unfolding of $\mathfrak{G}$, there is an N-unfolding relation $\mathsf{U} \subseteq \ell(\mathfrak{G}) \times \ell(\hat{\mathfrak{G}})$ such that $x\mathsf{U}\hat{x}$. We will break up the steps of Definition 5.2.1 into substeps that only add one or two formulas to one particular label y with $xR_{\mathfrak{G}}y$.

The only unfolding condition from Definition 6.4.5 that prevents U from making $\hat{\mathfrak{G}}$ an N-unfolding of $\mathfrak{G}'$ (and of $\mathfrak{G}''$) is $\mathsf{U1}$, and only with regard to labels $\hat{y}$ such that $y\mathsf{U}\hat{y}$ because y is the only label of $\mathfrak{G}$ with formulas added. We proceed depending on which step of Definition 5.2.1 was applied:

1) Suppose $\mathfrak{G}$,$x{:}A \wedge B^{\bullet}$ and the missing $x{:}A^{\bullet}$ and $x{:}B^{\bullet}$ were added to obtain $\mathfrak{G}'$. We use $N = n$ and we extend the premise $\hat{\mathfrak{G}}$ of $\mathcal{D}_n$ by the $\mathsf{cont}^{\bullet}$ and $\wedge^{\bullet}$ rules for each label $\hat{x}$ such that $x\mathsf{U}\hat{x}$ as follows:

$$\mathsf{cont}^{\bullet}\,\frac{\wedge^{\bullet}\,\dfrac{\mathcal{R}, \Gamma, \hat{x}{:}A \wedge B, \hat{x}{:}A, \hat{x}{:}B \Longrightarrow \Delta}{\mathcal{R}, \Gamma, \hat{x}{:}A \wedge B, \hat{x}{:}A \wedge B \Longrightarrow \Delta}}{\mathcal{R}, \Gamma, \hat{x}{:}A \wedge B \Longrightarrow \Delta}$$

It is easy to see that U makes $\hat{\mathfrak{G}}'$ a n-unfolding of $\mathfrak{G}'$ because ${}_{\hat{\mathfrak{G}}'}\hat{x} \sim {}_{\mathfrak{G}'}x$ whenever $x\mathsf{U}\hat{x}$.

2) The case when $\mathfrak{G}$,$x{:}A \vee B^{\circ}$ and $x{:}A^{\circ}$ and $x{:}B^{\circ}$ were added to obtain $\mathfrak{G}'$ is similar, with $\vee^{\circ}$ used instead of $\wedge^{\bullet}$.

3) Suppose $\mathfrak{G}$,$x{:}\Box A^{\bullet}$ and $\mathfrak{G}'$ is obtained by adding $y{:}A^{\bullet}$ and $y{:}\Box A^{\bullet}$ for only one label y with $xR_{\mathfrak{G}}y$. Note that $uR_{\hat{\mathfrak{G}}}u$ for each label $u \in \ell(\hat{\mathfrak{G}})$ due to $\hat{\mathfrak{G}}$ being proper. The sequence of rules depends on the position of x relative to y:

   (a) For each y that is not in the same cluster with x. We use $N = n$ and there exists $\hat{x}$ such that $x\mathsf{U}\hat{x}$. For each $\hat{y}$ such that $y\mathsf{U}\hat{y}$, we know that $\hat{x}R_{\hat{\mathfrak{G}}}\hat{y}$ by definition of unfolding. Hence, ito obtain $\hat{\mathfrak{G}}'$, we extend premise $\hat{\mathfrak{G}}$ of $\mathcal{D}_n$ by rules

$$\Box^{\bullet},4^{\bullet}\,\frac{\mathcal{R}, \hat{x}{\leq}\hat{x}, \hat{x}R\hat{y}, \Gamma, \hat{x}{:}\Box A, \hat{y}{:}\Box A, \hat{y}{:}A \Longrightarrow \Delta}{\mathcal{R}, \hat{x}{\leq}\hat{x}, \hat{x}R\hat{y}, \Gamma, \hat{x}{:}\Box A \Longrightarrow \Delta}$$

   for each label $\hat{y}$ such that $y\mathsf{U}\hat{y}$. This ensures that ${}_{\hat{\mathfrak{G}}'}\hat{y} \sim {}_{\mathfrak{G}'}y$ whenever $y\mathsf{U}\hat{y}$.

(b) For $x = y$, we use $N = n$ and we have each $\hat{x}$ such that $x\mathsf{U}\hat{x}$. To obtain $\mathfrak{G}'$, we extend premise $\hat{\mathfrak{G}}$ of D_n by rules

$$\Box^\bullet \frac{\mathcal{R}, \hat{x}R\hat{x}, \hat{x}{\leq}\hat{x}, \Gamma, \hat{x}{:}\Box A, \hat{x}{:}A \Longrightarrow \Delta}{\mathcal{R}, \hat{x}R\hat{x}, \hat{x}{\leq}\hat{x}, \Gamma, \hat{x}{:}\Box A \Longrightarrow \Delta}$$

(c) For $x \neq y$ that are in the same cluster $C = \{x_1, \ldots, x_k\}$ with $k > 1$, $x = x_i$, and $y = x_j$ for some $i, j = 1..k$. We use $N = n + 1$. To construct $\mathcal{D}'_n$, we apply the $\Box^\bullet$ and $\mathsf{4}^\bullet$ rules to $\hat{x}_{i1}$, i.e., the first unfolded copy of x, which we can call $\hat{x}_1$, and each of $\hat{x}_{j2}, \ldots, \hat{x}_{j,n+1}$, i.e., the 2nd, …, $(n+1)$th unfolded copies of y, which we can call $\hat{y}_2, \ldots, \hat{y}_{n+1}$, using the fact that $\hat{x}_1 R\hat{y}_2, \ldots, \hat{x}_1 R\hat{y}_{n+1}$ and doing the addition whenever $A^\bullet$ or $\Box A^\bullet$ has been added to y in $\mathfrak{G}$.

4) The case when $\mathfrak{G}$,$x{:}\Diamond A^\circ$, and $\mathfrak{G}'$ is obtained by adding $y{:}A^\circ$ and $y{:}\Diamond A^\circ$ for all labels y with $xR_{\mathfrak{G}}y$ where these formulas are not yet present is similar to the previous case, with $\Diamond^\circ$ and $\mathsf{4}^\circ$ instead of $\Box^\bullet$ and $\mathsf{4}^\bullet$.

5) We have $\mathfrak{G}$,$x{:}A \vee B^\bullet$ and $\mathfrak{G}$,$x{:}A^\bullet$ is added to obtain $\mathfrak{G}'$ and $\mathfrak{G}$,$x{:}B^\bullet$ is added to obtain $\mathfrak{G}''$. Here there are two subcases:

(a) If $\{x\}$ is a singleton, we use $N = n$. To construct $\mathcal{D}'_n$ from an n-unfolding $\hat{\mathfrak{G}}$ of $\mathfrak{G}$ in $\mathcal{D}_n$, for the unique $\hat{x}$ such that $x\mathsf{U}\hat{x}$ we will have the form $\mathcal{R}, \Gamma, \hat{x}{:}A \vee B \Longrightarrow \Delta$. Then we apply the $\mathsf{cont}^\bullet$-rule and $\vee^\bullet$-rule:

$$\mathsf{cont}^\bullet \frac{\vee^\bullet \dfrac{\mathcal{R}, \Gamma, \hat{x}{:}A \vee B, \hat{x}{:}A \Longrightarrow \Delta \quad \mathcal{R}, \Gamma, \hat{x}{:}A \vee B, \hat{x}{:}B \Longrightarrow \Delta}{\mathcal{R}, \Gamma, \hat{x}{:}A \vee B, \hat{x}{:}A \vee B \Longrightarrow \Delta}}{\mathcal{R}, \Gamma, \hat{x}{:}A \vee B \Longrightarrow \Delta}$$

for the unique label $\hat{x}$, such that the two premises are n-unfoldings of $\mathfrak{G}'$ and $\mathfrak{G}''$ respectively.

(b) If $\{x\}$ is in a non-singleton cluster, we use $N = 2n - 1$. Then in the $(2n-1)$-unfolding $\hat{\mathfrak{G}}$ of $\mathfrak{G}$ we have $\hat{x}_1, \ldots, \hat{x}_{2n-1}$ such that $\hat{\mathfrak{G}}$,$\hat{x}_j{:}A \vee B^\bullet$ for $j = 1, ..., 2n-1$. We can build a derivation using the $\vee^\bullet$-rule obtaining 2^{2n-1} premises of the form:

$$\begin{array}{c} \mathcal{R}, \Gamma, \hat{x}_1{:}A \vee B, \hat{x}_1{:}C_1, \\ \vdots \\ \hat{x}_{2n-1}{:}A \vee B, \hat{x}_{2n-1}{:}C_{2n-1} \Longrightarrow \Delta \end{array}$$

where C_j is either A or B, and each of these premises has either at least n times the formula A or at least n times the formula B in the places of $C_1, \ldots, C_{2n-1}$. For the other $n-1$ unfolding copies of x, we simply remove the U relations to them. This transforms each premise either into an n-unfolding of $\mathfrak{G}'$ or into an n-unfolding of $\mathfrak{G}''$. The derivations can be plugged into $\mathcal{D}_{2n-1}$ and we have thus constructed the derivation $\mathcal{D}'_n$.

6) We have $\mathfrak{G}$,$x{:}A \wedge B^\circ$ and $\mathfrak{G}$,$x{:}A^\circ$ is added to obtain $\mathfrak{G}'$ and $\mathfrak{G}$,$x{:}B^\circ$ is added to obtain $\mathfrak{G}''$. This is similar to the previous case, with $\wedge^\circ$ used instead of $\vee^\bullet$.

7) We have $\mathfrak{G}$,$x{:}A \supset B^\bullet$ and $\mathfrak{G}$,$x{:}A^\circ$ is added to obtain $\mathfrak{G}'$ and $\mathfrak{G}$,$x{:}B^\bullet$ is added to obtain $\mathfrak{G}''$. This is similar to the previous two cases, with $\supset^\bullet$ used instead of $\vee^\bullet$ or $\wedge^\circ$. $\square$

The next step is to show that $\overset{4}{\rightsquigarrow}_\Diamond$ preserves the property of being unfoldable. For this, we need to repeat parts of the proof. In order to simplify the process of repeating, we use the notion of *embedding*.

Definition 6.4.9 (Embedding)**.** Let $\mathfrak{G}$ and $\mathfrak{H}$ be proper sequents. An ***embedding*** $\mathsf{e}\colon \mathfrak{G} \to \mathfrak{H}$ is an injective function $\mathsf{e}\colon \ell(\mathfrak{G}) \to \ell(\mathfrak{H})$ obeying the following conditions for all $x, y \in \ell(\mathfrak{G})$:

(e1) $x \sim \mathsf{e}(x)$;

(e2) $xR_{\mathfrak{G}}y$ iff $\mathsf{e}(x)R_{\mathfrak{H}}\mathsf{e}(y)$;

(e3) $L_x \leq_{\mathfrak{G}} L_y$ iff $L_{\mathsf{e}(x)} \leq_{\mathfrak{H}} L_{\mathsf{e}(y)}$.

(e4) x has no past in $\mathfrak{G}$ iff $\mathsf{e}(x)$ has no past in $\mathfrak{H}$;

Proposition 6.4.10. *Embeddings are closed under composition and preserve unfoldings.*

Proof. First, we prove that a composition of two embeddings is an embedding. Let $\mathfrak{G}$, $\mathfrak{H}$, and $\mathfrak{J}$ be proper sequents. Let $\mathsf{e}\colon \mathfrak{G} \to \mathfrak{H}$ and $\mathsf{e}'\colon \mathfrak{H} \to \mathfrak{J}$ be embeddings from $\mathfrak{G}$ to $\mathfrak{H}$ and from $\mathfrak{H}$ to $\mathfrak{J}$, respectively. We need to show that $\mathsf{e}'\mathsf{e}$ is an embedding from $\mathfrak{G}$ to $\mathfrak{J}$, where $(e'e)(x) := \mathsf{e}'\Big(\mathsf{e}(x)\Big)$ for any $x \in \ell(\mathfrak{G})$. Since $\mathsf{e}\colon \ell(\mathfrak{G}) \to \ell(\mathfrak{H})$ and $\mathsf{e}'\colon \ell(\mathfrak{H}) \to \ell(\mathfrak{J})$ are injective functions, it is immediate that $\mathsf{e}'\mathsf{e}\colon \ell(\mathfrak{G}) \to \ell(\mathfrak{J})$ is an injective function. We have for all $x, y \in \ell(\mathfrak{G})$:

(e1) ${}_{\mathfrak{G}}x \sim {}_{\mathfrak{H}}\mathsf{e}(x) \sim {}_{\mathfrak{J}}\mathsf{e}'\Big(\mathsf{e}(x)\Big)$.

(e2) $xR_{\mathfrak{G}}y \quad$ iff $\quad \mathsf{e}(x)R_{\mathfrak{H}}\mathsf{e}(y) \quad$ iff $\quad \mathsf{e}'\Big(\mathsf{e}(x)\Big)R_{\mathfrak{J}}\mathsf{e}'\Big(\mathsf{e}(y)\Big)$.

(e3) $L_x \leq_{\mathfrak{G}} L_y \;$ iff $\; L_{\mathsf{e}(x)} \leq_{\mathfrak{H}} L_{\mathsf{e}(y)} \;$ iff $\; L_{\mathsf{e}'(\mathsf{e}(x))} \leq_{\mathfrak{J}} L_{\mathsf{e}'(\mathsf{e}(y))}$.

(e4) x has no past in $\mathfrak{G} \quad$ iff $\quad \mathsf{e}(x)$ has no past in $\mathfrak{H} \quad$ iff $\mathsf{e}'\Big(\mathsf{e}(x)\Big)$ has no past in $\mathfrak{J}$.

Secondly, let $\mathsf{e}\colon \hat{\mathfrak{G}} \to \hat{\mathfrak{H}}$ be an embedding and let $\mathsf{U} \subseteq \ell(\mathfrak{G}) \times \ell(\hat{\mathfrak{G}})$ be an n-unfolding relation witnessing that $\hat{\mathfrak{G}}$ is an n-unfolding of $\mathfrak{G}$. We show that binary relation $\mathsf{e}\mathsf{U} \subseteq \ell(\mathfrak{G}) \times \ell(\hat{\mathfrak{H}})$ witnesses that $\hat{\mathfrak{H}}$ is an n-unfolding of $\mathfrak{G}$, where

$$x(\mathsf{e}\mathsf{U})\hat{y} \qquad \Longleftrightarrow \qquad (\exists \hat{x})\Big(x\mathsf{U}\hat{x} \text{ and } \hat{y} = \mathsf{e}(\hat{x})\Big).$$

Let $x\,(\mathsf{e}\mathsf{U})\,\mathsf{e}(\hat{x})$ because $x\mathsf{U}\hat{x}$ and $x'\,(\mathsf{e}\mathsf{U})\,\mathsf{e}(\hat{x}')$ because $x'\mathsf{U}\hat{x}'$.

(U1) ${}_{\mathfrak{G}}x \sim {}_{\hat{\mathfrak{G}}}\hat{x}$ by (U1) for U and ${}_{\hat{\mathfrak{G}}}\hat{x} \sim {}_{\hat{\mathfrak{H}}}\mathsf{e}(\hat{x})$ by (e1) for e.

(U2) Let x and x' be from different clusters. Then $xR_{\mathfrak{G}}x'$ iff $\hat{x}R_{\hat{\mathfrak{G}}}\hat{x}'$ by (U2) for U and the latter iff $\mathsf{e}(\hat{x})R_{\hat{\mathfrak{H}}}\mathsf{e}(\hat{x}')$ by (e2) for e.

(U3) $L_x \leq_{\mathfrak{G}} L_{x'}$ iff $L_{\hat{x}} \leq_{\hat{\mathfrak{G}}} L_{\hat{x}'}$ by (U3) for U, and the latter iff $L_{\mathsf{e}(\hat{x})} \leq_{\hat{\mathfrak{H}}} L_{\mathsf{e}(\hat{x}')}$ by (e3) for e.

(U4) x has no past in $\mathfrak{G}$ iff $\hat{x}$ has no past in $\hat{\mathfrak{G}}$ by (U4) for U, and the latter iff $\mathsf{e}(\hat{x})$ has no past in $\hat{\mathfrak{H}}$ by (e4) for e.

(U5) If x forms a singleton cluster in $\mathfrak{G}$, then $\hat{x}$ is the only label in $\hat{\mathfrak{G}}$ such that $x\mathsf{U}\hat{x}$ by (U5) for U. Hence, $\mathsf{e}(\hat{x})$ is the only label in $\hat{\mathfrak{H}}$ such that $x\,(\mathsf{eU})\,\mathsf{e}(\hat{x})$.

(U6) If C is a non-singleton cluster in $\mathfrak{G}$ with $|C| = k \geq 2$, then by (U6) for U there are $x_i \in \ell(\mathfrak{G})$ and $\hat{x}_{i,j} \in \ell(\hat{\mathfrak{G}})$ for $i \in \{1,\ldots,k\}$ and $j \in \{1,\ldots,n\}$ $i = 1..k$ and $j = 1..n$ such that $C = \{x_1,\ldots,x_k\}$, and $x_i\mathsf{U}\hat{x}_{i,j}$ for $i = 1..k$, $j = 1..n$, and in $\hat{\mathfrak{G}}$

$$\hat{x}_{1,1}R\cdots R\hat{x}_{k,1}R\hat{x}_{1,2}R\cdots R\hat{x}_{k,2}R\cdots R\hat{x}_{1,n}R\cdots R\hat{x}_{k,n}.$$

In this case, $x_i\,(\mathsf{eU})\,\mathsf{e}(\hat{x}_{i,j})$ for $i = 1..k$, $j = 1..n$ and in $\hat{\mathfrak{H}}$ by (e2) for e

$$\mathsf{e}(\hat{x}_{1,1})\,R\cdots R\,\mathsf{e}(\hat{x}_{k,1})\,R\,\mathsf{e}(\hat{x}_{1,2})\,R\cdots R\,\mathsf{e}(\hat{x}_{k,2})\,R\cdots R\,\mathsf{e}(\hat{x}_{1,n})\,R\cdots R\,\mathsf{e}(\hat{x}_{k,n}).$$

Thus, $\mathsf{e}(\hat{x}_{i,j}) \in \ell(\hat{\mathfrak{H}})$ for $i = 1..k$, $j = 1..n$ provide the requisite n-unfoldings for the labels of C in $\hat{\mathfrak{H}}$.

(U7) If $\mathsf{e}(\hat{x}) = \mathsf{e}(\hat{x}')$, then $\hat{x} = \hat{x}'$ by the injectivity of e, and $x = x'$ by (U7) for U. □

Lemma 6.4.11 (Embedding Lemma)**.** *Assume we have a derivation $\mathcal{D}$ with a proper end-sequent $\mathfrak{G}$ and premises $\mathfrak{G}_1,\ldots,\mathfrak{G}_n$. If there is an embedding $\mathsf{e}\colon \mathfrak{G} \to \mathfrak{H}$ into a proper sequent $\mathfrak{H}$, then there is a derivation $\mathcal{D}'$ with conclusion $\mathfrak{H}$ and premises $\mathfrak{H}_1,\ldots,\mathfrak{H}_n$ such that there are embeddings $\mathsf{e}_i\colon \mathfrak{G}_i \to \mathfrak{H}_i$ for $i = 1..n$.*

Proof. Let us first consider the case where $\mathcal{D}$ consists of only one inference rule instance r. Let $\mathfrak{G}_i$ be a premise of r. If r does not add new labels, then $\ell(\mathfrak{G}_i) = \ell(\mathfrak{G})$, and we have $\ell(\mathfrak{H}_i) = \ell(\mathfrak{H})$ and $\mathsf{e}_i = \mathsf{e}$. Otherwise, if r adds new labels, then r is one of $\Diamond^\bullet$, $\supset^\circ$, or $\Box^\circ$, there is only one premise, and the new labels only occur in $\mathfrak{G}_1'$.

- r is $\Diamond^\bullet$: The rule applies to $x{:}\Diamond A^\bullet$ in $\mathfrak{G}$, creating a new label y, and we can apply the same rule to $\mathsf{e}(x){:}\Diamond A^\bullet$ in $\mathfrak{H}$, creating a new label y'. We can define $\mathsf{e}_1(y) = y'$, and for all other $u \in \ell(\mathfrak{G}_1)$ we define $\mathsf{e}_1(u) = \mathsf{e}(u)$.

- r is $\supset^\circ$ or $\Box^\circ$: The rule applies to $x{:}F^\circ$ in $\mathfrak{G}$, creating a new layer L' whose labels all occur only in $\mathfrak{G}_1' = \mathfrak{G}{\uparrow}_4^{x:F}$. We can apply the same rule to $\mathsf{e}(x){:}F^\circ$ in $\mathfrak{H}$, creating a new layer L'' and adding $\mathfrak{H}_1' = \mathfrak{H}{\uparrow}_4^{\mathsf{e}(x):F}$ to $\mathfrak{H}$. In general, L'' can contain more labels than L'. But e defines an embedding of L_x into $L_{\mathsf{e}(x)}$, which canonically defines a mapping $\mathsf{e}'\colon L' \to L''$ as follows: for $y' \in L'$, we define $\mathsf{e}'(y') = y'' \in L''$ iff either there is a $v \in L_x$ with $v{\leq_{\mathfrak{G}_1}}y'$ and $\mathsf{e}(v){\leq_{\mathfrak{H}_1}}y''$, or y' has no past in $\mathfrak{G}_1$ and y'' has no past in $\mathfrak{H}_1$. We can then define our embedding $\mathsf{e}_1\colon \mathfrak{G}_1 \to \mathfrak{H}_1$ as $\mathsf{e}_1(u) = \mathsf{e}'(u)$ if $u \in L'$ and $\mathsf{e}_1(u) = \mathsf{e}(u)$ otherwise.

- r is F_1 or F_2: The rule creates a new label y in $\mathfrak{G}$. We can apply the same rule in $\mathfrak{H}$ creating a new label y'. We define $\mathsf{e}_1(y) = y'$, and for all other $u \in \ell(\mathfrak{G}_1)$ we define $\mathsf{e}_1(u) = \mathsf{e}(u)$.

We can now prove the lemma for arbitrary $\mathcal{D}$ with a straightforward induction on the height of $\mathcal{D}$. □

Lemma 6.4.12. *Whenever for a semi-saturated sequent $\mathfrak{G}$, the algorithm in step 1) applies Option 1) of Definition 6.2.8 by substituting label x for label y in $\mathfrak{G}$ and applying the transitive closure of R to obtain $\mathfrak{G}'$ such that $\mathfrak{G} \overset{4}{\rightsquigarrow}_{\Diamond} \mathfrak{G}'$, label y forms a singleton cluster in $\mathfrak{G}$.*

Proof. Since the application of Option 1 means that y has no past in $\mathfrak{G}$, it follows that the algorithm created y as a singleton cluster (unless $y = r$, in which case it was present from the very beginning, but was a singleton cluster at first). It is not hard to check that every time the algorithm creates a new non-singleton cluster, either z have a past for all label z in the cluster or all labelled formulas $z{:}\Diamond B^{\bullet}$ become $\mathsf{IS4}$-happy for all label z in the cluster. Since $y{:}\Diamond A^{\bullet}$ has to be $\mathsf{IS4}$-unhappy and have no past to apply Option 1, label y is not part of a non-trivial cluster. □

Definition 6.4.13. For a label x of a sequent $\mathfrak{G}$ we define

$$\overrightarrow{x} := \{z \mid xR_{\mathfrak{G}}z\}.$$

Lemma 6.4.14. *If $\mathfrak{S}$ is F-unfoldable and $\mathfrak{G} \in \mathfrak{S}$ and $\mathfrak{G} \overset{4}{\rightsquigarrow}_{\Diamond} \mathfrak{G}'$ and $\mathfrak{S}' = \mathfrak{S} \setminus \{\mathfrak{G}\} \cup \{\mathfrak{G}'\}$, then $\mathfrak{S}'$ is F-unfoldable.*

Proof. Assume we have a set of sequents $\mathfrak{S}$ which is F-unfoldable and a sequent $\mathfrak{G} \in \mathfrak{S}$ and $\mathfrak{G} \overset{4}{\rightsquigarrow}_{\Diamond} \mathfrak{G}'$ and $\mathfrak{S}' = \mathfrak{S} \setminus \{\mathfrak{G}\} \cup \{\mathfrak{G}'\}$. For every $N \geq n$, we have an N-unfolding $\mathcal{D}_N$ of $\mathfrak{S}$, and we need to construct an n-unfolding $\mathcal{D}'_n$ of $\mathfrak{S}'$. According to Definition 6.2.8 there are two different cases:

1) In Option 1, we have that $\mathfrak{G}$,$y{:}\Diamond A^{\bullet}$ is $\mathsf{IS4}$-unhappy, where y forms a singleton cluster by Lemma 6.4.12, and we have some label x with $x \sim y$, $xR_{\mathfrak{G}}y$ and $x \neq y$, such that all $u \in \overrightarrow{x}$ have no past. Then $\mathfrak{G}'$ is obtained by substituting x for y and closing under transitivity and reflexivity. Consider any premise $\hat{\mathfrak{G}}$ of $\mathcal{D}_N$ that is an N-unfolding of $\mathfrak{G}$ via $\mathsf{U1}$. By $\mathsf{U5}$, there is a unique $\hat{y}$ such that $y\mathsf{U}\hat{y}$. The choice of N and the extension of $\mathcal{D}_N$ depend on the size of the cluster that x belongs to in $\mathfrak{G}$:

   a) If $\{x\}$ is a singleton cluster, we set $N = n$. There is a unique $\hat{x}$ such that $x\mathsf{U}\hat{x}$. We also have $\hat{x}R\hat{y}$ and since $\hat{x}$ has no past by assumption, neither has $\hat{x}$. It follows from tree-layeredness of $\hat{\mathfrak{G}}$ that all labels in $\overrightarrow{\hat{x}}$, including $\hat{y}$, have no past. Hence, all labels in $\overrightarrow{\hat{x}}$, with the possible exception of $\hat{x}$ itself, are created by rule $\Diamond^{\bullet}$. Let $\mathfrak{K}_1$ be the premise of the first (i.e., lowermost) instance of the rule $\Diamond^{\bullet}$ on the branch of $\mathcal{D}_n$ leading to $\hat{\mathfrak{G}}$ that creates a non-trivial child of $\hat{x}$, let us call this child u_1. Let $\hat{x}{:}\Diamond C^{\bullet}$ be the principal formula of this instance of $\Diamond^{\bullet}$. Then $\mathfrak{K}_1$,$u_1{:}C^{\bullet}$ is the only labelled formula in u_1. Since $\mathfrak{K}_1$,$\hat{x}{:}\Diamond C^{\bullet}$, also $\hat{\mathfrak{G}}$,$\hat{x}{:}\Diamond C^{\bullet}$ higher up the branch, hence, $\mathfrak{G}$,$x{:}\Diamond C^{\bullet}$ by $\mathsf{U1}$. Therefore, $\mathfrak{G}$,$y{:}\Diamond C^{\bullet}$ because $x \sim y$ and $\hat{\mathfrak{G}}$,$\hat{y}{:}\Diamond C^{\bullet}$. We now apply the rule $\Diamond^{\bullet}$ to $\hat{\mathfrak{G}}$ with $\hat{y}{:}\Diamond C^{\bullet}$ as its principal formula. We also apply $R\mathsf{rf}$ and $R\mathsf{tr}$ exhaustively we closed under reflexivity and transitivity of R. Let the premise of this rule be $\mathfrak{K}_2$ and u_2 be the new label created by this rule. Consider the following function $\mathsf{e}_1 \colon \ell(\mathfrak{K}_1) \to \ell(\mathfrak{K}_2)$:

$$\mathsf{e}_1 := \{(\hat{x}, \hat{y}), (u_1, u_2)\} \cup \{(w, w) \mid w \in \ell(\mathfrak{K}_1) \setminus \{\hat{x}, u_1\}\}.$$

It is easy to see that e_1 is an embedding. The derivation subtree rooted at $\mathfrak{K}_1$ can be applied to $\mathfrak{K}_2$ via this embedding resulting in an other derivation by Lemma 6.4.11 with all the premises of the extension being n-unfoldings of the same sequents from $\boldsymbol{\mathfrak{S}}$ as the respective premises of $\mathcal{D}_n$ by Proposition 6.4.10. In particular, for premise $\hat{\mathfrak{G}}$ of the subtree, embedding e_1 will be extended to an embedding into some premise $\hat{\mathfrak{G}}_2$ of the extended derivation with $\mathsf{e}_1(\hat{y}) = \hat{y}_2$ such that ${}_{\hat{\mathfrak{G}}_2}\hat{y}_2 \sim {}_{\hat{\mathfrak{G}}}\hat{y}$ (note that ${}_{\mathfrak{K}_2}\hat{y} \sim {}_{\hat{\mathfrak{G}}}\hat{y}$). By Proposition 6.4.10, we can construct an n-unfolding U_2 of $\mathfrak{G}$ into $\hat{\mathfrak{G}}_2$ such that $w\mathsf{U}_1\hat{w}$ iff $w\mathsf{U}_2\hat{w}$ for all labels $\hat{w} \in \ell(\mathfrak{K}_1) \setminus \{\hat{x}, u_1\} = \ell(\hat{\mathfrak{G}}) \setminus \{\hat{x}\}$, including for all non-trivial parents of $\hat{x}$. Furthermore, $x\mathsf{U}_2\hat{y}$ and, whenever $xR_{\mathfrak{G}}v$ and $v\mathsf{U}_2\hat{v}$ we have $\hat{y}R_{\hat{\mathfrak{G}}_2}\hat{v}$. In particular, $y\mathsf{U}_2\hat{y}_2$ for some $\hat{y}_2$ such that $\hat{y}R_{\hat{\mathfrak{G}}_2}\hat{y}_2$ and $\hat{\mathfrak{G}}_2{,}\hat{y}_2{:}\Diamond C^\bullet$.
We can apply $\Diamond^\bullet$ to obtain $\mathfrak{K}_3$ with a new label u_3, take an embedding

$$\mathsf{e}_2 := \{(\hat{y}, \hat{y}_2), (u_2, u_3)\} \cup \{(w, w) \mid w \in \ell(\mathfrak{K}_2) \setminus \{\hat{y}, u_2\}\}.$$

from $\mathfrak{K}_2$ into $\mathfrak{K}_3$ and repeat the process.

After $n-1$ such repetitions, we will get a premise $\hat{\mathfrak{G}}_{n+1}$ and an n-unfolding U_{n+1} of $\mathfrak{G}$ into $\hat{\mathfrak{G}}_{n+1}$ such that $w\mathsf{U}_1\hat{w}$ iff $w\mathsf{U}_{n+1}\hat{w}$ for all labels $\hat{w} \in \ell(\hat{\mathfrak{G}}) \setminus \{\hat{x}\}$, including for all non-trivial parents of $\hat{x}$. Furthermore, $x\mathsf{U}_{n+1}\hat{y}_n$ and, whenever $xR_{\mathfrak{G}}v$ and $v\mathsf{U}_{n+1}\hat{v}$ we have $\hat{y}_nR_{\hat{\mathfrak{G}}_{n+1}}\hat{v}$. In particular, $y\mathsf{U}_{n+1}\hat{y}_{n+1}$ for some y_{n+1} such that $\hat{y}_nR_{\hat{\mathfrak{G}}_{n+1}}\hat{y}_{n+1}$. Once again, all other premises of this derivation expanded n times are n-unfoldings of sequents from $\boldsymbol{\mathfrak{S}}$. It remains to turn $\hat{\mathfrak{G}}_{n+1}$ into an n-unfolding of $\mathfrak{G}'$. We do this by modifying U_{n+1} for the newly created cluster $C_x = \{u \neq y \mid xR_{\mathfrak{G}}uR_{\mathfrak{G}}y\}$ in $\mathfrak{G}'$ and keeping unfoldings of all other labels of $\mathfrak{G}$ as in U_{n+1}. For each label $u \in C_x$ and each $j = 1..n$ there is at least one label $\hat{u}_j$ such that $u\mathsf{U}_j\hat{u}_j$ and $\hat{y}_{j-1}R_{\hat{\mathfrak{G}}_{n+1}}\hat{u}_jR_{\hat{\mathfrak{G}}_{n+1}}\hat{y}_j$. These labels $\hat{u}_1, \ldots, \hat{u}_n$ for each $u \in C_x$ can, thus, be taken as the requisite n-unfolding copies of C_x. Indeed, whenever $wR_{\mathfrak{G}}x$ and $w\mathsf{U}_{n+1}\hat{w}$ for some $w \neq x$, as discussed, we have $\hat{w}R_{\hat{\mathfrak{G}}_{n+1}}\hat{x}R_{\hat{\mathfrak{G}}_{n+1}}\hat{u}_j$. On the other hand, whenever $xR_{\mathfrak{G}}v$ and $v\mathsf{U}_{n+1}\hat{v}$ for some $v \notin C_x \cup \{y\}$, as discussed, we have $\hat{u}_jR_{\hat{\mathfrak{G}}_{n+1}}\hat{y}_nR_{\hat{\mathfrak{G}}_{n+1}}\hat{v}$. Note that children of x that are not in C_x in $\mathfrak{G}'$ must be unfolded into labels that are children of all $\hat{u}_j$, which is why we needed to create $n+1$ successful unfoldings rather than n: the first n-unfoldings are used to create n unfolded copies of C_x whereas the last $(n{+}1)$th provides unfoldings for children of x that are not in its cluster.

b) If $\{x\}$ belongs to a non-singleton cluster C_x, we use a similar expansion to the case of singleton clusters just discussed. Hence, we will only outline the differences. Firstly, instead of $\mathcal{D}_n$ we start with $\mathcal{D}_N$ for $N = n+2$. The fact that $N \geq n$ guarantees n-unfoldings for all premises other than $\mathfrak{G}$ and for all clusters of $\mathfrak{G}$ other than the newly created C'_x that contains C_x and all labels in between C_x and y, excluding y.
Among the $N \geq 2$ unfolding copies of x in $\hat{\mathfrak{G}}$, use the penultimate, $(N{-}1)$th unfolding copy as $\hat{x}$ and the unique unfolding of y as $\hat{y}$. For these $\hat{x}$ and $\hat{y}$ perform the expansion as in the case for a singleton cluster. The only difference to that case is how the final unfolding U_{n+1} is to be modified to account for the newly created cluster C'_x. For all labels outside of C_x, the construction remains the same. For each label $u \in C_x$, its last, Nth unfolding copy from U_1 is among labels repeated in the segment (note that the same need not be true about the $(N{-}1)$th unfolding copy, which is why we had to start one unfolding copy earlier). Hence, this Nth copy is now repeated $n{+}1$ times, and the first n of these are taken as new unfoldings of u.

2) In Option 2, we have $\mathfrak{G}$,y:$\Diamond A^\bullet$ IS4-unhappy, and $\mathfrak{G}'$ is obtained from $\mathfrak{G}$ by adding yRz, $z{\leq}z$, and z:$A^\bullet$ for some fresh z, and closing R under reflexivity and transitivity. Note that z is a singleton cluster. We provide an abbreviated description how to expand $\hat{\mathfrak{G}}$ that is an N-unfolding of $\mathfrak{G}$ into an n-unfolding of $\mathfrak{G}'$. The procedure depends on the size of the cluster that y belongs to in $\mathfrak{G}$:

   a) If $\{y\}$ is a singleton cluster, it has a unique unfolding $\hat{y}$, such that $y\mathsf{U}\hat{y}$. We use $N = n$ and obtain $\mathcal{D}'_n$ from $\mathcal{D}_n$ by applying to $\hat{y}$:$\Diamond A^\bullet$ the $\mathsf{cont}^\bullet$-rule, then creating a fresh label $\hat{z}$ by the application of $\Diamond^\bullet$-rule, applying reflexivity and transitivity rules as needed, and adding $z\mathsf{U}\hat{z}$ to the unfolding:

$$\mathit{R}\mathsf{rf}{+}{\leq}\mathsf{rf}\ \frac{\mathcal{R}, \hat{y}R\hat{y}, \hat{z}R\hat{z}, \hat{y}R\hat{z}, \hat{z}{\leq}\hat{z}, \Gamma, \hat{y}{:}\Diamond A, \hat{z}{:}A \Longrightarrow \Delta}{\Diamond^\bullet\ \dfrac{\mathcal{R}, \hat{y}R\hat{z}, \Gamma, \hat{y}{:}\Diamond A, \hat{z}{:}A \Longrightarrow \Delta}{\mathsf{cont}^\bullet\ \dfrac{\mathcal{R}, \Gamma, \hat{y}{:}\Diamond A, \hat{y}{:}\Diamond A \Longrightarrow \Delta}{\mathcal{R}, \Gamma, \hat{y}{:}\Diamond A \Longrightarrow \Delta}}}\ \hat{z}\text{ fresh}$$

   b) If $\{y\}$ is in a non-singleton cluster C_y, we use $N = n + 1$ so that there must be $n + 1$ unfolded copies $\hat{y}_1, \ldots, \hat{y}_{n+1}$ in $\hat{\mathfrak{G}}$. We apply $\Diamond^\bullet$-rule to $\hat{y}_{n+1}$:$\Diamond A^\bullet$ creating label $\hat{z}$ (is made the unique unfolding of z), applying $\mathit{R}\mathsf{rf}$ and $\mathit{R}\mathsf{tr}$ as needed, adding $z\mathsf{U}\hat{z}$ to the unfolding and removing all U relations to the $(n+1)$th copy of C_y in $\hat{\mathfrak{G}}$. The removal of the extra unfoldings in the last item is necessary to make sure that $\hat{z}$ is R-accessible from each of the unfoldings of C_y. This is also the reason why $\mathsf{cont}^\bullet$ is not necessary to apply to $\hat{y}_{n+1}$ to preserve the $\Diamond^\bullet$ as $\hat{y}_{n+1}$ need not remain equivalent to y. □

Lemma 6.4.15. *The rewrite relation* $\overset{4}{\rightrightarrows}_\Diamond$ *is sound.*

Proof. This follows immediately from Lemmas 6.4.14 and 6.4.8. □

Lemma 6.4.16. *If* x:A *is an allowed formula in* $\mathfrak{G}$ *and* $\mathfrak{G} \in \mathfrak{S}$ *and* $\mathfrak{S}$ *is* F*-unfoldable, then* $\mathfrak{S}' = \mathfrak{S} \setminus \{\mathfrak{G}\} \cup \{\mathfrak{G}{\uparrow}_4^{x:A}\}$ *is* F*-unfoldable.*

Proof. Let $\mathfrak{G} \in \mathfrak{S}$ and $\mathfrak{G}{\uparrow}_4^{x:A}$ be the sequent obtained from Construction 6.2.15 for x:A an allowed formula. By hypothesis (in order to preserve (U5) and (U6)), we have a derivation $\mathcal{D}_{2n+1}$ of $\mathfrak{G}_0(F)$ where each premise is an $(2n+1)$-unfolding of a sequent in $\mathfrak{S}$. Let $\mathfrak{H}$ be an $(2n+1)$-unfolding of $\mathfrak{G}$. We construct a derivation with conclusion $\mathfrak{H}$, where we apply the following steps for the IS4-unhappy $\mathfrak{G}$,x:A°:

1. If x is a singleton-cluster in $\mathfrak{G}$, then apply the rule corresponding to A° ($\supset^\circ$ or $\Box^\circ$) in the corresponding label in $\mathfrak{H}$. Otherwise, if x is in a cluster C with $|C| > 1$, then apply that rule in the $(n+1)$th repetition of the cluster C in $\mathfrak{H}$. In each case we get a unique label $\hat{x}$ with $x \leq \hat{x}$.

2. Apply F_1 and F_2 to create a copy $\hat{L}$ of the whole layer L of x in $\mathfrak{H}$.

3. Exhaustively apply $\mathsf{mon}^\bullet$, $\mathit{R}\mathsf{rf}$, $\mathit{R}\mathsf{tr}$, ${\leq}\mathsf{rf}$, ${\leq}\mathsf{tr}$, $4^\bullet$, 4°.

4. For every lifted cluster that does not contain x the additional $n+1$ copies in the unfolding are discarded. But when x is in a cluster, then $\hat{L}$ contains 2 copies of that cluster (see Figure 6.2), whose n repetitions are given to us by the copies $1 \cdots n$ and $(n+1) \cdots 2(n+1)$ of our lifting. In the nth copy, only the lifting of x is part of the unfolding relation.

Then the resulting sequent $\mathfrak{H}'$ is an n-unfolding of $\mathfrak{G}\uparrow_4^{x:A}$, and the derivation with premise $\mathfrak{H}'$ and conclusion $\mathfrak{H}$ can be plugged into $\mathcal{D}_{2n+1}$ wherever $\mathfrak{H}$ occurs as a leaf. This yields the desired n-unfolding of $\mathfrak{G}'$. □

In the following Lemma, we prove that the rewrite relation $\twoheadrightarrow_{\circlearrowright}$ is sound on IS4-saturated set of sequents:

Lemma 6.4.17. *If $\mathfrak{G} \twoheadrightarrow_{\circlearrowright} \hat{\mathfrak{G}}$ and $\mathfrak{G}$ is IS4-saturated and F-unfoldable, then $\hat{\mathfrak{G}}$ is F-unfoldable.*

Proof. Assume we have $\mathfrak{G} \rightsquigarrow_{\circlearrowright} \mathfrak{G}'$ and $\mathfrak{G} \in \mathfrak{G}$ saturated and $\mathfrak{G}$ is F-unfoldable, and $\mathfrak{G}' = \mathfrak{G} \setminus \{\mathfrak{G}\} \cup \{\mathfrak{G}'\}$. We want to construct an n-unfolding for $\mathfrak{G}'$. $\mathfrak{G}'$ is obtained from $\mathfrak{G}$ by closing an unhappy triangle loop. Let $L_1, L_2, L', C_1, C_2, p_1, s, t$ as in Definitions 6.2.17 and 6.2.18. We also have $p_2 \in C_2$ and $p_2 \sim p_1$, that we will use in the proof.

If s and t are in the same cluster, then every n-unfolding of $\mathfrak{G}$ is also an n-unfolding of $\mathfrak{G}'$. Now, assume that s and t are not in the same cluster.

We start from an $(n+2)$-unfolding $\mathcal{D}_{n+2}$ of $\mathfrak{G}$, and let $\hat{\mathfrak{G}}$ be a premise of $\mathcal{D}_{n+2}$ that is an unfolding of $\mathfrak{G}$. Let $\hat{L}_1, \hat{L}_2, \hat{L}'$ be the unfoldings of L_1, L_2, L', respectively.

We now let $\mathcal{T}$ be the subtree of $\mathcal{D}_{n+2}$ rooted at the sequent where $\hat{L}'$ first occurs. Let $\hat{\mathfrak{G}}_0$ be that sequent, let r be the instance of the inference rule (an $\Box^\circ$ or $\supset^\circ$) that introduced $\hat{L}'$, let $\mathfrak{H}$ be the conclusion of that rule instance, and let $\hat{p}_1$ be the label in $\mathfrak{H}$ containing the formula to which that rule was applied. Then, we have $p_1 \mathsf{U} \hat{p}_1$. Let $\hat{p}_2$ be a label in $\hat{\mathfrak{G}}$ with $p_2 \mathsf{U} \hat{p}_2$. If p_2 is a singleton, then $\hat{p}_2$ is uniquely defined. Otherwise, we pick for $\hat{p}_2$ the kth repetition of the cluster C_2, where k is the repetion of C_1 in which $\hat{p}_1$ occurred. We can apply the same rule r to $\hat{p}_2$ to obtain the layer $\hat{L}''$. Let the new sequent be $\hat{\mathfrak{G}}_1$.

We also let $\hat{p}_1'$ be the future of $\hat{p}_1$ in $\hat{L}'$, and let $\hat{p}_2'$ be the future of $\hat{p}_2$ in $\hat{L}''$. This allows us to define an embedding $\mathsf{e}\colon \hat{\mathfrak{G}}_0 \to \hat{\mathfrak{G}}_1$ acting like the identity on all layers, except $\hat{L}'$ which is embedded into $\hat{L}''$ mapping each label to its unique future, except for $\overrightarrow{\hat{p}_1}$ which is embedded into $\overrightarrow{\hat{p}_2}$. We are now going to apply the Embedding Lemma 6.4.11 to repeat the proof tree $\mathcal{T}$. More precisely, let $\mathfrak{H}_1, \ldots, \mathfrak{H}_m$ be the leaves of $\mathcal{T}$ and let $\mathfrak{H}_1, \ldots, \mathfrak{H}_l$ for some $l \leq m$ be the leaves of $\mathcal{T}$ that are N-unfoldings of $\mathfrak{G}$ (and $\hat{\mathfrak{G}}$ is among them). Each of $\mathfrak{H}_i$ has a layer $\hat{L}_{2,i}$ that is an N-unfolding of L_2, and that has the labels $\hat{s}_i$, $\hat{t}_i$, and $\hat{p}_{2,i}$, corresponding to s, t, and p_2 in the unfolding (if s is in a cluster, we pick for $\hat{s}_i$ the first repetition; if t is in a cluster, we pick for $\hat{t}_i$ the last repetition).

We proceed now similarly to the proof of Lemma 6.4.14 by plugging in a copy of the subtree $\mathcal{T}$ at each leaf $\mathfrak{H}_1, \ldots, \mathfrak{H}_l$. Note that whenever a new layer is created and the previous one is lifted using the $\supset^\circ$ or $\Box^\circ$-rules, also the gap between the old and the new triangle is lifted (see Figure 6.9). We call $\hat{L}_{3,i}$ the new created layer (which is a copy of $\hat{L}_{2,i}$ with an additional part $\hat{L}^*_{3,i}$ that is a copy of labels in $\hat{L}_{2,i}$ but that have no counterpart in $\hat{L}_1$. If a rule instance in $\mathcal{T}$ is adding a formula to one of the labels u with $s\, R_{\mathfrak{h}_i}\, u\, R_{\mathfrak{h}_i}\, t$ then we need do the same to the corresponding label in $\hat{L}^*_{3,i}$. Formally, this can be achieved via a second embedding for each i, applying again Lemma 6.4.11. As in the previous Lemma 6.4.14, we now repeat this construction n times, so that we obtain our desired n-unfolding of $\mathfrak{G}'$. □

Lemma 6.4.18 (Unfolding Lemma)**.** *All sets $\mathfrak{G}_i$ of sequents generated by the algorithm from Figure 6.4 are F-unfoldable.*

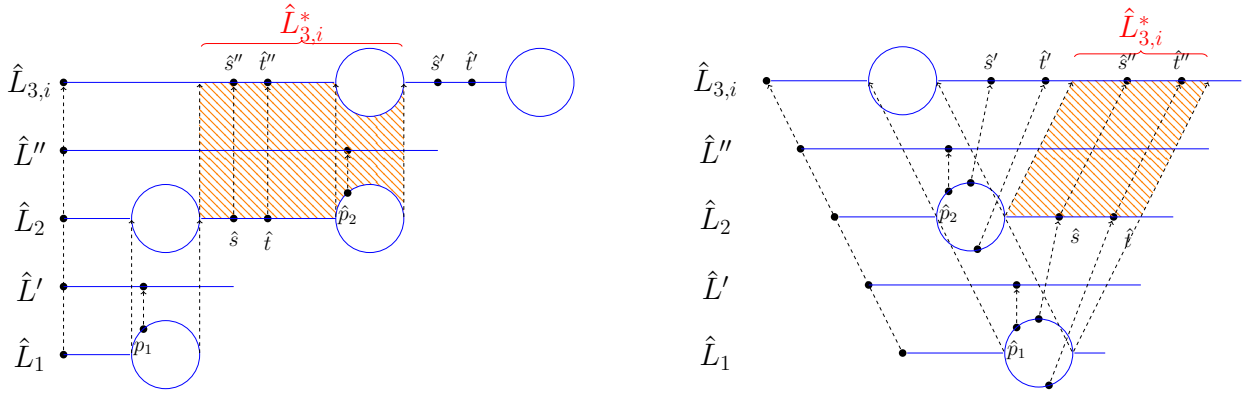


Figure 6.9: Left: Unfolding of an unhappy R-triangle loop. Right: Unfolding of an unhappy U-triangle loop

Proof. The set $\mathfrak{S}'_0 = \{\mathfrak{G}_0(F)\}$ is trivially F-unfoldable. By the previous lemmas, this property is preserved by all the steps that are used to construct the sequence $\mathfrak{S}_0, \mathfrak{S}_1, \cdots$ □

Theorem 6.4.19. *If the algorithm shown in Figure 6.4 terminates in Step 2, then the formula F is a theorem of* IS4.

Proof. The algorithm produces a sequence $\mathfrak{S}_0, \mathfrak{S}_1, \ldots$ of sets of sequents which are F-unfoldable by Lemma 6.4.18. Hence, also the set $\mathfrak{S}_k$ that caused the algorithm to terminate because it is axiomatic is F-unfoldable. In particular, we have a derivation with conclusion $\mathfrak{G}_0(F)$ where all premises are 1-unfoldings of elements of $\mathfrak{S}_k$, which are all axiomatic. Therefore we can use instances of the id and $\bot^\bullet$ rules to give a complete proof of $\mathfrak{G}_0(F)$ in labIS4$'_{\leq}$ and append it with rules Rrf and $\leq$rf at the bottom to get the proof of $\Longrightarrow r{:}F$. By Corollary 6.4.1 and 4.6.3, we have that F is a theorem of IS4. □

6.5 Termination

We have already established that every step in our algorithm terminates: semi-saturation was proved in Chapter 5 in Lemma 5.2.2, b), 4-saturation in Lemma 6.2.12 and loop saturation in Lemma 6.2.20,b). It remains to show that we cannot run through the main loop forever as we have done in Chapter 5, Section 5.5 showing that the number of distinct possible layers is finite and we will eventually terminate either in Step 2 (finding a proof) or in Step 3.b) (finding a countermodel).

We also have seen by Lemma 5.5.1 that the size of a label occuring in a sequent of some $\mathfrak{S}_i$ is at most n, and that there are 2^n many equivalence classes of labels with respect to $\sim$. We now need to check that the size of clusters are also finite. The ***size*** $|C|$ of a cluster C is the number of labels in C and we get the following result:

Lemma 6.5.1. *The size of a cluster occurring in a sequent of some $\mathfrak{S}_i$ is at most 2^n. And there are $2^{2^n} - 1$ many equivalence classes of (non-empty) clusters with respect to $\sim$.*

Proof. We shall show that, by construction, all labels in a cluster are different with respect to $\sim$. There are two ways of introducing non-singleton clusters:

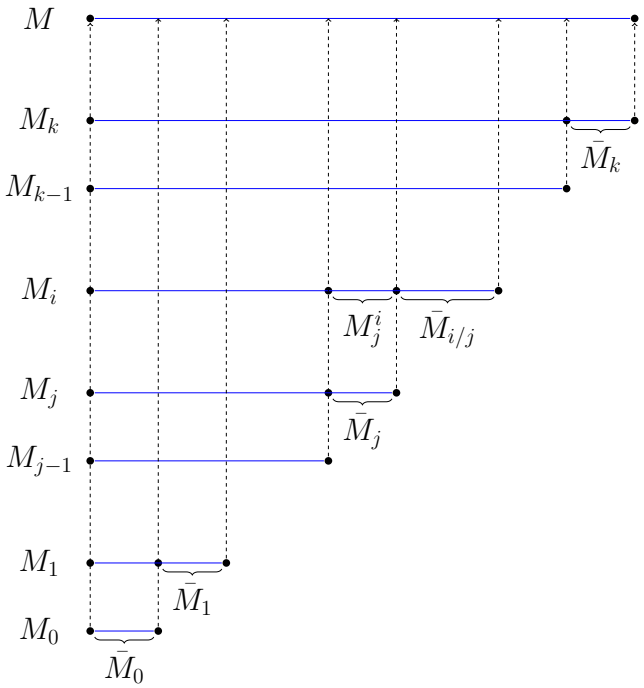


Figure 6.10: A branch and its pasts

(i) via IS4-unhappy $\Diamond^\bullet$-formulas (Option 1 of Definition 6.2.8): if there are equivalent labels in such a cluster, the cluster would have been created at an earlier step of the algorithm.

(ii) via unhappy R-triangle or U-triangle loops (Definitions 6.2.17, 6.2.18 and 6.2.19): if at creation the chain from s to t contains duplicates, then they are collapsed immediately afterwards, as they are also valid choices of s and t, fulfilling the same unhappy triangle conditions.

Hence, when $\rightsquigarrow\!\!\!\!\!\!\circlearrowright$ terminates, in all new clusters all labels are different with respect to $\sim$. This limits the size of clusters to 2^n, and we have at most $2^{2^n}-1$ different equivalence classes of non-empty clusters which can occur in $\mathfrak{G}_i$. □

The algorithm visits IS4-saturated sequents. A layer L in such a sequent is a tree of clusters. Let M be a branch in L, i.e., a sequence $C_1, C_2, \ldots, C_l$ of clusters from the root to a leaf. The ***length*** $|M|$ of M is the sum $|C_1| + |C_2| + \cdots + |C_l|$.

Lemma 6.5.2. *The length of a branch in a layer in a sequent in a set $\mathfrak{G}_i$ is bounded, and the bound is determined by F.*

Proof. A branch M' is a *past* of a branch M in a sequent $\mathfrak{G}$, if for all $x' \in M'$ there is a $x \in M$ with $x' \leq_{\mathfrak{G}} x$. In this case we write $M' \leq_{\mathfrak{G}} M$. For a branch M we write $\check{M}$ for the labels in M that have a past, and $\bar{M}$ for the ones that do not have a past. Then $M = \check{M} \cup \bar{M}$ and $\check{M} \cap \bar{M} = \emptyset$. We now consider a branch M in a layer L such that whenever $L \leq L'$ for some layer L', then $L = L'$; and we consider all its pasts, i.e., branches M' with $M' \leq_{\mathfrak{G}} M$.

Let $M_0, M_1, \ldots, M_k$ be all the pasts of M for which $\bar{M}_i \neq \emptyset$, and such that $M_i \leq_{\mathfrak{G}} M_j$ whenever $0 \leq i \leq j \leq k$ (see Figure 6.10). We have $\left|\bar{M}_i\right| \leq 2^n + 1$ for all i because every label in $\bar{M}_i$ is created by Option 2 of Definition 6.2.8, in correspondence to an $\mathsf{IS4}$-unhappy $\Diamond^\bullet$-formula (except for the first label in $\bar{M}_i$, which could be created by layer lifting in correspondence to an unhappy $\Box^\circ$ formula (Construction 6.2.15), or could be the initial label created in Step 0 of the algorithm). But in any case, after at most 2^n such steps, we encounter a label which is equivalent to a previous one (see also the proof of Lemma 6.2.12).

Now let $M^i_j = \{y \in M_i \mid \exists x \in \bar{M}_j.\, x \leq_{\mathfrak{G}} y\}$ be the set of futures of $\bar{M}_j$ in M_i for some $j < i$ (see Figure 6.10). Because of the duplication of clusters in the layer lifting (Construction 6.2.13), we can have $\left|M^i_j\right| > \left|\bar{M}_j\right|$. We can restrict the size of M^i_j because there is only a limited amount of times a cluster can be duplicated before a U-triangle loops is encountered. Then, the length of the branch from s and t before and after the repetition of the cluster is again bound by $2 \cdot (2^n + 1)$, because otherwise the U-triangle loop would be unhappy. (We have to take $2^n + 1$ twice because the suricata label is not allowed to occur between s and t). Since the size of a cluster is bound by 2^n, and the number of clusters in $\bar{M}_j$ is bound by 2^n, the size of M^i_j is bound by $2 \cdot 2 \cdot 2^n \cdot 2^n \cdot 2^n = 2^{3n+2}$.

Let us now put a bound to k. Very naively, we see an R-triangle loop after $k_{\max} = 2^n \cdot 2^{2^n}$ (as we need to repeat the cluster C_1, see Definition 6.2.17 and Lemma 6.5.1). Let M_j be the branch that sees a repetition of M_i. Define $\bar{M}_{i/j}$ (for $j < i$) to be the set of labels in M_i that do not have a past in M_j (see Figure 6.10). Then the R-triangle loop is unhappy if $\left|\bar{M}_{i/j}\right| \geq 2 \cdot 2^n = 2^{n+1}$ (same argument as above). Since $2^{3n+2} > 2^{n+1}$, we have that $|M| \leq k_{\max} \cdot \left|M^i_j\right| = 2^n \cdot 2^{2^n} \cdot 2^{3n+2} = 2^{2^n+4n+2}$.

Observe that as soon we observe an unhappy triangle loop, the size of the newly created cluster is $|C| \leq 2^n$. □

Theorem 6.5.3. *The proof search algorithm given in Figure 6.4 is terminating for* $\mathsf{IS4}$.

Proof. As we have done in Theorem 5.5.3, but using Lemma 6.5.2. □

6.6 Conclusion

In this chapter we have solved a problem which has been opened for almost 30 years [21]: we present a search algorithm (based on the one from Chapter 5) to prove decidability of $\mathsf{IS4}$. Our solution has two key ingredients:

First, the use of the fully labelled sequent system with relational atoms for both binary relations enabled us to give a proof system that has only invertible rules and also gives a closer correspondence between sequents and models.

Second, although the identification of labels during proof search to realize loops is a priori unsound, however, under the right circumstances, we can preserve soundness if we organize the proof search in a certain systematic way.

We conjecture that the same method can also be applied to $\mathsf{IK4}$, which is $\mathsf{IS4}$ without the t-axiom and which is the other logic in the intuitionistic version of the $\mathsf{S5}$-cube for which decidability is an open problem. In fact, the overall argument is the same, but in many definitions and proof arguments, there would be subtle differences due to the absence of reflexivity. For this reason, a treatement of the logic $\mathsf{IK4}$ is part of our future work.

Chapter 7

Subatomic logic for modalities

In this chapter we present a work in progress in the formalism of deep inference using the methodology of *subatomic proof theory* for the linear fragment of classical modal logic K. The subatomic methodology treats atoms like binary connectives, and thus allows a more uniform cut-elimination procedure in deep inference. In this chapter we extend this idea to modal logics. This means that also the unary modalities are treated as binary connectives. We present a subatomic proof system to capture the linear fragment of classical modal logic and we prove cut-elimination via splitting for this fragment.

In Section 7.1 we start with an introduction to the formalism of deep inference and the properties that deep inference systems provide. In Section 7.2 we present the preliminaries of subatomic logic, making emphasize in classical propositional logic [1]. In Section 7.3 we introduce subatomic formulae for the linear fragment of classical modal logic and a subatomic proof system capturing the modalities. In the same section we proceed showing cut-elimination via splitting and, as a result of this proof, we prove the admissibility of the up-rules which are the cuts. For a more detailed presentation of subatomic logic we refer to the PhD thesis of Aler Tubella [1].

7.1 From deep inference to subatomic logic

We have been studying in the previous chapters formalisms for modal logics in which the proofs progress by manipulating the outermost connectives of the formula trees. We are now interested in the study of the formalism of *deep inference* [25, 9] which abandons the importance of the main connective. In the last twenty years, the development of this new paradigm has increased. It allows the rewriting of formulas deep inside an arbitrary context and not just along their main connective as it is done, for example, in sequent calculus. The first such formalism is the *calculus of structures* introduced by Guglielmi [26, 25]. It derives its name from the fact that there is no distinction between sequents and formulas, but there is a unique syntactic structure which can be seen as an equivalence class of formulas modulo associativity and commutativity and unit equations that are sometimes imposed on sequents (in the previous chapters the comma is associative and concatenation with the empty sequent does not change a sequent).

Figure 7.1 shows the deep inference system $\mathsf{SK}^{\mathsf{K}}\mathsf{S}$ for the modal logic K (whose syntax and semantics were defined in Chapter 2). It is a variation of the system presented in [28]. The rule $\mathsf{k}_{\downarrow}$ resembles the axiom of distributivity k and $\mathsf{k}_{\uparrow}$ is its dual (if we remove the rules

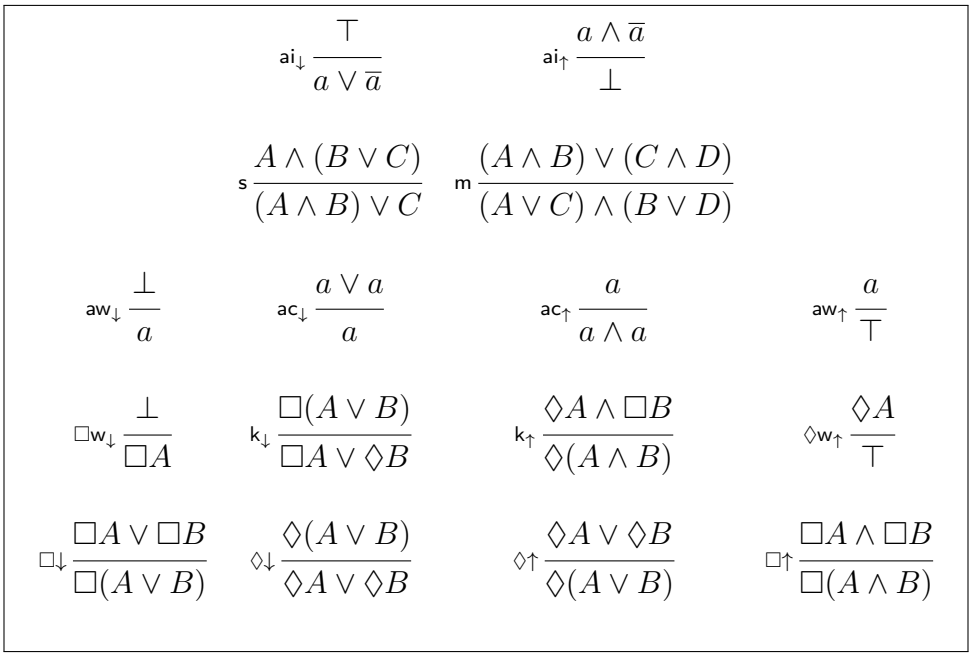


Figure 7.1: System $\mathsf{SK^KS}$

for the modalities we obtain the system SKS for classical propositional logic introduced by Brünnler and Tiu [9]). The rules in $\mathsf{SK^KS}$ are applied modulo the following equations:

$$\begin{array}{llllll} A \wedge (B \wedge C) = (A \wedge B) \wedge C & A \wedge B = B \wedge A & A \wedge \top = A & \Box\top = \top & \top \vee \top = \top \\ A \vee (B \vee C) = (A \vee B) \vee C & A \vee B = B \vee A & A \vee \bot = A & \Diamond\bot = \bot & \bot \wedge \bot = \bot \end{array}$$

That system (and all the deep inference systems) has another property: all rules are *local*. This is one important feature of the formalism of deep inference, in the sense that determining whether an application of the rule is correct we do not need to inspect arbitrarily big formulae. This is achieved by turning all structural rules such as contraction and cut rules into their atomic versions. This provides a surprising regularity in the inference rule schemes: it can be observed that in most deep inference systems all inference rules besides the atomic ones can be expressed as

$$\frac{(A\alpha B)\beta(C\alpha' D)}{(A\beta C)\alpha(B\beta' D)},$$

where A, B, C, D are formulae and $\alpha, \beta, \alpha', \beta'$ are connectives.

Atomicity is only possible in systems where we can apply rules deep. In the deep inference system $\mathsf{SK^KS}$, the identity axiom can be reduced to an atomic form as it is possible in sequent calculus. However, by duality, we can do the same for the cut rule, which is not possible in the sequent calculus. Furthermore, we can also transform contraction and weakening in an atomic form, which is also impossible in the sequent calculus and we obtain the rules $\mathsf{ac}_{\downarrow}, \mathsf{ac}_{\uparrow}, \mathsf{aw}_{\downarrow}, \mathsf{aw}_{\uparrow}$ from Figure 7.1.

Lemma 7.1.1. *The general weakening* $\mathsf{w}_{\downarrow}\dfrac{\bot}{A}$ *and co-weakeaning* $\mathsf{w}_{\uparrow}\dfrac{A}{\top}$ *are admissible in* $\mathsf{SK^KS}$.

Proof. We proceed by induction on the size of the principal formula of the rule:

- If $A = B \wedge C$, then we can do the following replacement:

$$\mathsf{w}_\downarrow \dfrac{\bot}{B \wedge C} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\bot}{=}}{\mathsf{w}_\downarrow \dfrac{\bot \wedge \bot}{\bot \wedge C}}}{\mathsf{w}_\downarrow \dfrac{}{B \wedge C}} \qquad\qquad \mathsf{w}_\uparrow \dfrac{B \wedge C}{\top} \quad \rightsquigarrow \quad \dfrac{\mathsf{w}_\uparrow \dfrac{B \wedge C}{B \wedge \top}}{= \dfrac{B}{\mathsf{w}_\uparrow \dfrac{}{\top}}}$$

- If $A = B \vee C$, then the proof is similar as the previous one.
- If $A = \Box B$, then it follows from the inference rule $\Box\mathsf{w}_\downarrow$ and from the equation $\Box\top = \top$.
- If $A = \Diamond B$, then it follows from the equation $\Diamond\bot = \bot$ and from the inference rule $\Diamond\mathsf{w}_\uparrow$.
- If $A = a$ it follows from the inference rules $\mathsf{aw}_\downarrow$ and $\mathsf{aw}_\uparrow$. □

Furthermore, in deep inference systems every rule has its *dual*. We indicate the dual of a rule $\rho_\downarrow$ by $\rho_\uparrow$. The concept of dual rule very intuitively corresponds to the idea of the contrapositive of an implication. Since the premise and conclusion of a deep inference rule are single formulas, dual rules can be very easily defined through the negation of formulas. We can define the negation (or dual) of a formula inductively, using De Morgan equivalences. Two rules are dual to each other if one can be obtained from the other by exchanging premise and conclusion and negating them: for example the identity rule $\mathsf{ai}_\downarrow$ and the cut rule $\mathsf{ai}_\uparrow$ of Figure 7.1. Some rules are said to be *self-dual*, i.e., the rule and its dual are identical. For example, the medial and the switch rules are self-dual (rules m and s in Figure 7.1 respectively).

Deep inference gives more flexibility in the design of inference rules. However, cut-elimination for deep inference systems is more involved than for traditional sequent style systems. In particular, for modal logics, no cut-elimination proof that is internal to deep inference has been given so far. There are many different cut-elimination techniques in the deep inference literature (but none of them for modal logics), exploiting different aspects of the proof systems they work on. A particular methodology does however stand out for its generality and its the one we are interested to study in this thesis: *cut-elimination via splitting* [26]. Even though this proof has to be redone for every proof system anew, there is a certain repeating pattern. In order to formalize this pattern and to obtain a general cut-elimination method that works for many proof systems at the same time, the method of *subatomic proof theory* [1] has been developed. The basic idea is to treat atoms as binary connectives, leading to a uniform shape of all inference rules (including the atomic rules). This enormously reduces the number of cases in the case analysis for cut-elimination.

A proof of cut-elimination via splitting usually consists of two parts. Only the second one is the actual splitting and needs a "linear" system, i.e., one without weakening and contraction. To remove weakening and contraction, the first part of the cut-elimination performs a decomposition [26] or cycle elimination [2].

7.2 Subatomic logic for classical propositional logic

In this section we present some preliminaries for subatomic logic, making emphasis on classical propositional logic. This work can be found in Aler Tubella's PhD thesis [1].

Subatomic formulae are built by freely composing constants by connectives and atoms. For example, these are two subatomic formulae for classical propositional logic:

$$A \equiv ((\bot a \top) \vee \top) \wedge (\top b \bot) \qquad B \equiv ((\top b \bot) \wedge \top) \vee \bot$$

The main idea is to interpret $\bot a \top$ as a positive occurrence of the atom a, and $\top a \bot$ as a negative occurrence of the same atom, denoted by $\overline{a}$. Intuitively, we can view subatomic formulae as a superposition of truth values. For example, $\bot a \top$ is the superposition of the two possible assignments for the atom a, and $\top a \bot$ is the superposition of the possible assignments for $\overline{a}$: if we read the value on the left of the atom we assign $\bot$ to a and $\top$ to $\overline{a}$, and if we read the one on the right we assign $\top$ to a and $\bot$ to $\overline{a}$. In order to understand the interpretation we are using and to translate the subatomic formula into the standard formula in the given logic, we define an interpretation map function.

Definition 7.2.1 (Subatomic formulae)**.** Let $\mathcal{U}$ be a denumerable set of constants whose elements are denoted by $u, v, w, ...$ Let $\mathcal{R}$ be a denumerable partially ordered set of connectives whose elements are denoted by $\alpha, \beta, \gamma, ...$ The set $\mathcal{F}$ of subatomic formulae (or SA formulae) contains terms defined by the grammar

$$\mathcal{F} ::= \mathcal{U} \mid \mathcal{F}\ \mathcal{R}\ \mathcal{F}$$

A *(formula) context* $K\{\}...\{\}$ is a formula where some subformulae are substituted by holes; $K\{A_1\}...\{A_n\}$ denotes a formula where the n holes in $K\{\}...\{\}$ have been filled with $A_1, ..., A_n$.

The expression $A \equiv B$ means that the formulae A and B are syntactically equal. We omit parentheses when there is no ambiguity. In $K\{A \alpha B\}$ we say that the subformulae of A and B are in the scope of α.

Example 7.2.1 (Subatomic formulae for classical logic)**.** The set $\mathcal{F}_{\mathtt{cl}}$ of subatomic formulae for classical logic is given by the set of constants $\mathcal{U} = \{\bot, \top\}$ and the set of relations $\mathcal{R} = \{\wedge, \vee\} \cup \mathcal{A}$ where $\mathcal{A}$ is a denumerable set of atoms, denoted by $a, b, ...$ with $\mathcal{A} \cap \{\wedge, \vee\} = \emptyset$.

Definition 7.2.2 (Interpretation map function)**.** Let $\mathcal{G}$ be the set of formulae of a logic L, and let $\mathcal{F}$ be the set of subatomic formulae with constants $\mathcal{U}$ and connectives $\mathcal{R}$. A surjective partial function $\mathtt{I} : \mathcal{F} \to \mathcal{G}$ is called an *interpretation map*. The domain of definition of $\mathtt{I}$ is the set of *interpretable formulae* and is denoted by $\mathcal{F}^i$.

For $A \in \mathcal{F}$, we say that R is the interpretation of A and A is a *representation* of R if $R \equiv \mathtt{I}(A)$.

The extension of the notion of interpretability to contexts arises from the following definition:

Definition 7.2.3 (Interpretation of contexts)**.** We say that a context $S\{\}$ is *interpretable* if $S\{A\}$ is interpretable for every interpretable A.

An interpretation for the set of subatomic formulae for classical logic defined in Example 7.2.1 is given by considering the assignments:

$$\begin{array}{cc} \mathtt{I}(\top) = \top & \mathtt{I}(\bot) = \bot \\ \forall a \in A.\ \mathtt{I}(\bot a \bot) = \bot & \forall a \in A.\ \mathtt{I}(\top a \top) = \top \\ \forall a \in A.\ \mathtt{I}(\bot a \top) = a & \forall a \in A.\ \mathtt{I}(\top a \bot) = \overline{a} \\ \mathtt{I}(A \vee B) = \mathtt{I}(A) \vee \mathtt{I}(B) & \mathtt{I}(A \wedge B) = \mathtt{I}(A) \wedge \mathtt{I}(B) \end{array}$$

where $A, B \in \mathcal{F}^i$, and extending it in such a way that AaB is interpretable if and only if $A = u, B = v$ with $u, v \in \{\bot, \top\}$ and then $\mathtt{I}(AaB) \equiv \mathtt{I}(u\ a\ v)$.

For example, we can now interpret the subatomic formula $A \equiv ((\bot a \top) \vee \top) \wedge (\top b \bot)$ as $\mathtt{I}(A) = (a \vee \top) \wedge \overline{b}$. Note that the set $\mathcal{F}^i$ of interpretable formulae is composed by all formulae equal to a formula where an atom does not occur in the scope of another atom. Every other formula is not interpretable, such as $B \equiv ((\top b \bot) \wedge \top) a \bot$.

Definition 7.2.4 (Negation and Equational theory)**.** We define *negation* as a pair of involutive maps $\overline{\cdot} : \mathcal{R} \to \mathcal{R}$ and $\overline{\cdot} : \mathcal{U} \to \mathcal{U}$. We define the *negation map on formulae* as the map inductively defined by setting $\overline{A \alpha B} := \overline{A} \overline{\alpha} \overline{B}$.

We define an *equational theory* $=$ on $\mathcal{F}$ as the minimal equivalence relation closed under negation (if $A = B$, then $\overline{A} = \overline{B}$) and under context (if $A = B$, then $K\{A\} = K\{B\}$ for any context $K\{\}$) defined by a set of axioms of the form:

1. (Associativity of α) $\forall A, B, C \in \mathcal{F}.\ (A\ \alpha\ B)\ \alpha\ C = A\ \alpha\ (B\ \alpha\ C)$
2. (Commutativity of α) $\forall A, B \in \mathcal{F}.\ A\ \alpha\ B = B\ \alpha\ A$
3. (Unit of α) $\forall A \in \mathcal{F}.\ A\ \alpha\ u_\alpha = A = u_\alpha\ \alpha\ A$, for a fixed $u_\alpha \in \mathcal{U}$

If there is an axiom of the form (1) for α, we say that α is associative. If there is an axiom of the form (2) for α, we say that α is commutative. If there is an axiom of the form (3) for α we say that α is unitary, and we call u_α the unit of α.

Example 7.2.2 (Equational theory for classical logic)**.** For the set of subatomic formulae for classical logic $\mathcal{F}_{\mathtt{cl}}$ defined in example 7.2.1, the negation is defined as follows:

$$\begin{array}{c} \overline{\wedge} := \vee; \\ \overline{a} := a \text{ for all } a \in A; \\ \overline{\bot} := \top \end{array}$$

We define the equational theory $=$ on $\mathcal{F}_{\mathtt{cl}}$ as the minimal equivalence relation closed under negation defined by:

For all $A, B, C \in \mathcal{F}$:

$$\begin{array}{cc} (A \wedge B) \wedge C = A \wedge (B \wedge C) & (A \vee B) \vee C = A \vee (B \vee C) \\ A \wedge B = B \wedge A & A \vee B = B \vee A \\ A \wedge \top = A & A \vee \bot = A \\ \bot \wedge \bot = \bot & \top \vee \top = \top \\ \forall a \in A. \bot a \bot = \bot & \forall a \in A. \top a \top = \top \end{array}$$

$$\mathsf{sai}_{\downarrow}\frac{(A \vee B)\ a\ (C \vee D)}{(A\ a\ C) \vee (B\ a\ D)} \qquad \mathsf{sai}_{\uparrow}\frac{(A\ a\ B) \wedge (C\ a\ D)}{(A \wedge C)\ a\ (B \wedge D)}$$

$$\wedge_{\downarrow}\frac{(A \vee B) \wedge (C \vee D)}{(A \wedge C) \vee (B \vee D)} \qquad \vee_{\uparrow}\frac{(A \vee B) \wedge (C \wedge D)}{(A \wedge C) \vee (B \wedge D)}$$

$$\mathsf{m}\frac{(A \wedge B) \vee (C \wedge D)}{(A \vee C)\ a\ (B \vee D)}$$

$$\mathsf{sac}_{\downarrow}\frac{(A\ a\ B) \vee (C\ a\ D)}{(A \vee C)\ a\ (B \vee D)} \qquad \mathsf{sac}_{\uparrow}\frac{(A \wedge B)\ a\ (C \wedge D)}{(A\ a\ C) \wedge (B\ a\ D)}$$

Figure 7.2: System SAKS

We can now introduce the subatomic proof system for classical propositional logic SAKS [1] presented in Figure 7.2. As it was mentioned in the previous section, in deep inference systems all the inference rules besides the atomic ones can be expressed with the same shape. However, the useful properties of subatomic formulae become apparent when we extend the principle to atomic inference rules. Let us consider, for example, the usual contraction and identity rule for an atom (rules $\mathsf{ac}_{\downarrow}$ and $\mathsf{ai}_{\downarrow}$ in Figure 7.1). We could obtain these rules subatomically by reading $\bot a \top$ as a and $\top a \bot$ as $\overline{a}$, as follows:

$$\mathsf{ac}_{\downarrow}\frac{a \vee a}{a} \ \rightarrow\ \mathsf{sac}_{\downarrow}\frac{(\bot a \top) \vee (\bot a \top)}{(\bot \vee \bot)a(\top \vee \top)} \qquad \mathsf{ai}_{\downarrow}\frac{\top}{a \vee \overline{a}} \ \rightarrow\ \mathsf{sai}_{\downarrow}\frac{(\bot \vee \top)a(\bot \vee \top)}{(\bot a \top) \vee (\top a \bot)}$$

where $\mathsf{sac}_{\downarrow}$ and $\mathsf{sai}_{\downarrow}$ are subatomic inference rules in system SAKS.

7.3 Subatomic modal logic

As it was mentioned before, the subatomic methodology treats atoms as binary connectives. The question now arises: *how can we capture the modalities in this new methodology given that they are unary connectives?*. To address this and to capture $\Box$ and $\Diamond$, we introduce a new operator $\boxtimes$. Then, we define the subatomic formulae for a specific fragment of classical modal logic, which is its linear fragment. We refer to it as LinK and we define its subatomic formulae as follows:

Definition 7.3.1 (Subatomic formulae for LinK)**.** The set $\mathcal{F}_{\mathtt{lml}}$ of subatomic formulae for classical modal logic is given by the set of constants $\mathcal{U} = \{\bot, \top\}$ and the set of relations $\mathcal{R} = \{\boxtimes, \wedge, \vee\} \cup \mathcal{A}$ where $\mathcal{A}$ is a countable set of atoms, denoted by $a, b, ...$ with $\mathcal{A} \cap \{\boxtimes, \wedge, \vee\} = \emptyset$.

In order to interpret this subatomic formulae we need to define the interpretation map as we have seen for classical propositional logic in the previous section.

Definition 7.3.2 (Interpretation map for LinK)**.** Let $\mathcal{G}$ be the set of formulae of classical modal logic, and let $\mathcal{F}$ be the set of subatomic formulae with constants $\mathcal{U}$ and connectives $\mathcal{R}$. We define an interpretation function $\mathtt{I} : \mathcal{F} \to \mathcal{G}$ for the set of subatomic formulae for LinK as follows:

$$\begin{array}{cc} \mathtt{I}(\top) = \top & \mathtt{I}(\bot) = \bot \\ \forall a \in A.\ \mathtt{I}(\bot a \bot) = \bot & \forall a \in A.\ \mathtt{I}(\top a \top) = \top \\ \forall a \in A.\ \mathtt{I}(\bot a \top) = a & \forall a \in A.\ \mathtt{I}(\top a \bot) = \overline{a} \\ \mathtt{I}(A \vee B) = \mathtt{I}(A) \vee \mathtt{I}(B) & \mathtt{I}(A \wedge B) = \mathtt{I}(A) \wedge \mathtt{I}(B) \\ \mathtt{I}(\top \boxtimes A) = \Box \mathtt{I}(A) & \mathtt{I}(\bot \boxtimes A) = \Diamond \mathtt{I}(A) \end{array}$$

where A, B are interpretable formulae, and extending it in such a way that AaB is interpretable if and only if $A = u, B = v$ with $u, v \in \{\bot, \top\}$ and then $\mathtt{I}(AaB) \equiv \mathtt{I}(u\ a\ v)$. The formula $A \boxtimes B$ is only intepretable if and only if $A = u$ with $u \in \{\top, \bot\}$.

Remark 7.3.3. Since we are working on the linear fragment of classical modal logic, observe that the formulae that are not interpretable are not only those equal to a formula where an atom occurs in the scope of another atom, but also those where a formula made up of units not equal to $\top$ or $\bot$ occurs in the scope of an atom or in the scope of $\boxtimes$, such as $(\top \vee \top)a\bot$ or $(\top \vee \top) \boxtimes A$.

Remark 7.3.4. Observe that the interpretation defined for LinK is the interpretation defined for classical propositional logic by adding the two conditions for the new operator $\boxtimes$.

The interpretation of $\bot a \top$ and $\top a \bot$ are as in the classical case. The first one is a positive occurrence of the atom a, and the second one, a negative occurrence $\overline{a}$ of the same atom. An example of a subatomic formula for LinK is

$$A \equiv ((\bot \boxtimes (\bot a \top)) \vee (\top b \bot)) \wedge (\top \boxtimes (\bot a \top))$$

and it is interpreted as $\mathtt{I}(A) = (\Diamond a \vee \bar{b}) \wedge \Box a$ which is a standard classical modal formula.

7.3.1 The subatomic system

In Figure 7.3, we present a subatomic proof system to capture the linear fragment of classical modal logic which it is called $\mathsf{SAK^KS}$. In particular, it is an extension of the system for classical logic SAKS [1] of Figure 7.2 with the inference rules to capture $\Box$ and $\Diamond$, and without the contraction rules since we are working with the linear part.

The rules in $\mathsf{SAK^KS}$ are applied modulo associativity and commutativity of $\wedge$ and $\vee$, and the following equations:[1]

$$\begin{array}{lll} A \wedge \top = A & \forall a \in A. \bot a \bot = \bot & \top \boxtimes \top = \top \\ A \vee \bot = A & \forall a \in A. \top a \top = \top & \bot \boxtimes \bot = \bot \end{array}$$

In the same way as it was done for the other rules in the subatomic system for classical propositional logic, we can now obtain the subatomic version of the rules for the modalities $\mathsf{k}_\downarrow$ and $\mathsf{k}_\uparrow$ from the deep inference system for K of Figure 7.1 as follows:

[1] Observe that we lack both $\top \vee \top = \top$ and $\bot \wedge \bot = \bot$.

$$\mathsf{sai}_{\downarrow}\frac{(A\vee B)\ a\ (C\vee D)}{(A\ a\ C)\vee(B\ a\ D)}\qquad \mathsf{sai}_{\uparrow}\frac{(A\ a\ B)\wedge(C\ a\ D)}{(A\wedge C)\ a\ (B\wedge D)}$$

$$\wedge_{\downarrow}\frac{(A\vee B)\wedge(C\vee D)}{(A\wedge C)\vee(B\vee D)}\qquad \wedge_{\uparrow}\frac{(A\vee B)\wedge(C\wedge D)}{(A\wedge C)\vee(B\wedge D)}$$

$$\mathsf{m}\frac{(A\wedge B)\vee(C\wedge D)}{(A\vee C)\ \wedge\ (B\vee D)}$$

$$\mathsf{sak}_{\downarrow}\frac{(A\vee C)\boxtimes(B\vee D)}{(A\boxtimes B)\vee(C\boxtimes D)}\qquad \mathsf{sak}_{\uparrow}\frac{(A\boxtimes C)\wedge(B\boxtimes D)}{(A\wedge B)\boxtimes(C\wedge D)}$$

$$\mathsf{m}_{\boxtimes\downarrow}\frac{(A\boxtimes B)\vee(C\boxtimes D)}{(A\vee C)\boxtimes(B\vee D)}\qquad \mathsf{m}_{\boxtimes\uparrow}\frac{(A\wedge B)\boxtimes(C\wedge D)}{(A\boxtimes C)\wedge(B\boxtimes D)}$$

Figure 7.3: System $\mathsf{SAK^KS}$

$$\mathsf{k}_{\downarrow}\frac{\Box(A\vee B)}{\Box A\vee\Diamond B}\ \rightsquigarrow\ \mathsf{sak}_{\downarrow}\frac{(\top\vee\bot)\boxtimes(A\vee B)}{(\top\boxtimes A)\vee(\bot\boxtimes B)}\qquad \mathsf{k}_{\uparrow}\frac{\Diamond A\wedge\Box B}{\Diamond(A\wedge B)}\ \rightsquigarrow\ \mathsf{sak}_{\uparrow}\frac{(\bot\boxtimes A)\wedge(\top\boxtimes B)}{(\bot\wedge\top)\boxtimes(A\wedge B)}$$

We can now prove that the subatomic system $\mathsf{SAK^KS}$ is a conservative extension for the linear fragment of the deep inference system $\mathsf{SK^KS}$. In other words, we have the following statement:

Theorem 7.3.5. *For every interpretable* $\mathsf{SAK^KS}$ *derivation with premise* P *and conclusion* C*, there is a derivation in the linear fragment of* $\mathsf{SK^KS}$ *with premise* $\mathtt{I}(P)$ *and conclusion* $\mathtt{I}(C)$*.*

Proof. We have to prove that every interpretable assignment in the inference rules of the subatomic system $\mathsf{SAK^KS}$ has a corresponding derivation in the linear fragment of the deep inference system $\mathsf{SK^KS}$. We follow the interpretation defined in Definition 7.3.2 and Remark 7.3.3.

For the rule $\mathsf{sai}_{\downarrow}$ we have the following interpretable assignments:

$$\mathsf{sai}_{\downarrow}\frac{(\bot\vee\bot)\ a\ (\bot\vee\bot)}{(\bot a\bot)\vee(\bot a\bot)}\xrightarrow{\mathtt{I}}\frac{\bot}{\bot\vee\bot}$$

$$\mathsf{sai}_{\downarrow}\frac{(\bot\vee\top)\ a\ (\top\vee\bot)}{(\bot a\top)\vee(\top a\bot)}\xrightarrow{\mathtt{I}}\frac{\top}{a\vee\overline{a}}\qquad \mathsf{sai}_{\downarrow}\frac{(\top\vee\bot)\ a\ (\bot\vee\top)}{(\top a\bot)\vee(\bot a\top)}\xrightarrow{\mathtt{I}}\frac{\top}{\overline{a}\vee a}$$

$$\mathsf{sai}_{\downarrow}\frac{(\bot\vee\top)\ a\ (\bot\vee\top)}{(\bot a\bot)\vee(\top a\top)}\xrightarrow{\mathtt{I}}\frac{\top}{\bot\vee\top}\qquad \mathsf{sai}_{\downarrow}\frac{(\top\vee\bot)\ a\ (\top\vee\bot)}{(\top a\top)\vee(\bot a\bot)}\xrightarrow{\mathtt{I}}\frac{\top}{\top\vee\bot}$$

$$\mathsf{sai}_{\downarrow}\frac{(\bot\vee\top)\ a\ (\bot\vee\bot)}{(\bot a\bot)\vee(\top a\bot)}\xrightarrow{\mathtt{I}}\frac{\overline{a}}{\bot\vee\overline{a}}\qquad \mathsf{sai}_{\downarrow}\frac{(\bot\vee\bot)\ a\ (\bot\vee\top)}{(\bot a\bot)\vee(\bot a\top)}\xrightarrow{\mathtt{I}}\frac{a}{\bot\vee a}$$

$$\mathsf{sai}_{\downarrow}\frac{(\top\vee\bot)\ a\ (\bot\vee\bot)}{(\top a\bot)\vee(\bot a\bot)}\xrightarrow{\mathtt{I}}\frac{\overline{a}}{\overline{a}\vee\bot}\qquad\qquad \mathsf{sai}_{\downarrow}\frac{(\bot\vee\bot)\ a\ (\top\vee\bot)}{(\bot a\top)\vee(\bot a\bot)}\xrightarrow{\mathtt{I}}\frac{a}{a\vee\bot}$$

It is straightforward that there is a $\mathsf{SK}^{\mathsf{K}}\mathsf{S}$ derivation with the same premise and conclusion as the interpretation. The rule $\mathsf{sai}_{\uparrow}$ is dual.

For rule $\mathsf{sak}_{\downarrow}$ we have the following interpretable assignments following Definition 7.3.2 and Remark 7.3.3:

$$\mathsf{sak}_{\downarrow}\frac{(\top\vee\bot)\ \boxtimes\ (A\vee B)}{(\top\boxtimes A)\vee(\bot\boxtimes B)}\xrightarrow{\mathtt{I}}\frac{\Box(A\vee B)}{\Box A\vee\Diamond B}$$ which corresponds to the rule $\mathsf{k}_{\downarrow}$ in $\mathsf{SK}^{\mathsf{K}}\mathsf{S}$

$$\mathsf{sak}_{\downarrow}\frac{(\bot\vee\top)\ \boxtimes\ (A\vee B)}{(\bot\boxtimes A)\vee(\top\boxtimes B)}\xrightarrow{\mathtt{I}}\frac{\Box(A\vee B)}{\Diamond A\vee\Box B}$$ which corresponds to the rule $\mathsf{k}_{\downarrow}$ in $\mathsf{SK}^{\mathsf{K}}\mathsf{S}$

$$\mathsf{sak}_{\downarrow}\frac{(\bot\vee\bot)\ \boxtimes\ (A\vee B)}{(\bot\boxtimes A)\vee(\bot\boxtimes B)}\xrightarrow{\mathtt{I}}\frac{\Diamond(A\vee B)}{\Diamond A\vee\Diamond B}$$ which corresponds to the rule $\Diamond\downarrow$ in $\mathsf{SK}^{\mathsf{K}}\mathsf{S}$.

The rule $\mathsf{sak}_{\uparrow}$ is dual and we get the interpretable assignments as follows:

$$\mathsf{sak}_{\uparrow}\frac{(\top\boxtimes A)\ \wedge\ (\bot\boxtimes B)}{(\top\wedge\bot)\boxtimes(A\wedge B)}\xrightarrow{\mathtt{I}}\frac{\Box A\wedge\Diamond B}{\Diamond(A\wedge B)}$$ which corresponds to the rule $\mathsf{k}_{\uparrow}$ in $\mathsf{SK}^{\mathsf{K}}\mathsf{S}$

$$\mathsf{sak}_{\uparrow}\frac{(\bot\boxtimes A)\ \wedge\ (\top\boxtimes B)}{(\bot\wedge\top)\boxtimes(A\wedge B)}\xrightarrow{\mathtt{I}}\frac{\Diamond A\wedge\Box B}{\Diamond(A\wedge B)}$$ which corresponds to the rule $\mathsf{k}_{\uparrow}$ in $\mathsf{SK}^{\mathsf{K}}\mathsf{S}$

$$\mathsf{sak}_{\uparrow}\frac{(\top\boxtimes A)\ \wedge\ (\top\boxtimes B)}{(\top\wedge\top)\boxtimes(A\wedge B)}\xrightarrow{\mathtt{I}}\frac{\Box A\wedge\Box B}{\Box(A\wedge B)}$$ which corresponds to the rule $\Box\uparrow$ in $\mathsf{SK}^{\mathsf{K}}\mathsf{S}$.

For the rest of the rules in $\mathsf{SAK}^{\mathsf{K}}\mathsf{S}$ the proof is similar. Furthermore, we can check every interpretable instance of a rule inside the scope of an atom which is necessarily an instance where the premise and conclusion are interpreted as constants. The only such instances corresponds to a rule $\dfrac{u}{u}$ with $u\in\{\bot,\top\}$, and therefore every interpretable instance of a rule inside the scope of an atom trivially corresponds to a derivation in $\mathsf{SK}^{\mathsf{K}}\mathsf{S}$. □

7.3.2 Splitting

The standard syntactic method for proving cut-elimination in the sequent calculus is to permute the cut rule upwards in the proof, while decomposing the cut formula along its main connective, and so inductively reduce the cut rank. However, in systems formulated using deep inference this method cannot be applied, as derivations can be constructed in a more flexible way than in the sequent calculus. For this reason, the splitting technique has been developed in the literature of deep inference.

The cut-elimination proof via splitting [1] is with respect to splittable systems. These are the systems where the scope of connectives only increases reading from bottom to top. We then prove cut-elimination via splitting for the splittable system obtained from $\mathsf{SAK}^{\mathsf{K}}\mathsf{S}$. We called this system $\mathsf{SAK}^{\mathsf{K}}\mathsf{S}_{\mathsf{sp}}$ which is composed by the rules $\mathsf{sai}_{\downarrow}$, $\wedge_{\downarrow}$ and $\mathsf{sak}_{\downarrow}$ of Figure 7.3.

Definition 7.3.6. We define $=_\vee$ as the equivalence relation on formulae defined by the axioms for the associativity, commutativity and unit of $\vee$.

Definition 7.3.7 (Tame)**.** We say that an interpretable derivation ϕ in a subatomic proof system SA is *tame* if the only instances of rules in the scope of an atom are equality rules.

Definition 7.3.8 (Length of a proof)**.** Given a proof ϕ in $\mathsf{SAK^K S_{sp}}$, we define the length of ϕ as the number of instances of inference rules in ϕ different from the equality rules for the associativity, commutativity and unit of $\vee$. We denote it by $|\phi|_\vee$.

Theorem 7.3.9 (Splitting)**.** *For all formulae A, B, C and $\alpha \in \{\wedge, \boxtimes, a\}$, if there is a proof ϕ of $(A \alpha B) \vee C$ in $\mathsf{SAK^K S_{sp}}$, then there exist Q_1, Q_2 and derivations*

$$\dfrac{\dfrac{Q_1 \;\overline{\alpha}\; Q_2}{\dfrac{}{C}\,\psi}}{C}\,, \qquad \dfrac{\dfrac{}{A \;\vee\; Q_1}\,\phi_1}{A \;\vee\; Q_1} \quad \textit{and} \quad \dfrac{\dfrac{}{B \;\vee\; Q_2}\,\phi_2}{B \;\vee\; Q_2}$$

such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi|_\vee$. Furthermore, if ϕ is tame, then ϕ_1, ϕ_2 and ψ are tame.

Proof. 1. Given a proof ϕ of $(A \boxtimes B) \vee C$ in $\mathsf{SAK^K S_{sp}}$, we proceed by induction on $|\phi|_\vee$.

(a) If $|\phi|_\vee =_\vee 0$, then $(A \boxtimes B) \vee C =_\vee \top$. Then, either:

- $A =_\vee B =_\vee \top, C =_\vee \bot$ and we take

$$\psi \equiv \dfrac{\dfrac{\bot \boxtimes \bot}{\bot}\,=}{\dfrac{}{C}\,=_\vee}\,, \quad \phi_1 \equiv \dfrac{\dfrac{}{\boxed{=_\vee \dfrac{\top}{A}} \;\vee\; \bot}\,=_\vee}{\top}\,, \; \phi_2 \equiv \dfrac{\dfrac{}{\boxed{=_\vee \dfrac{\top}{B}} \vee \bot}\,=_\vee}{\top}$$

  with $|\phi_1|_\vee = |\phi_2|_\vee = 0$; or,

- $A =_\vee B =_\vee C =_\vee \top$ and we take

$$\psi \equiv \dfrac{\dfrac{\top \boxtimes \top}{\top}\,=}{\dfrac{}{C}\,=_\vee}\,, \quad \phi_1 \equiv \dfrac{\dfrac{}{\boxed{=_\vee \dfrac{\top}{A}} \;\vee\; \top}\,=_\vee}{\top}\,, \; \phi_2 \equiv \dfrac{\dfrac{}{\boxed{=_\vee \dfrac{\top}{B}} \vee \top}\,=_\vee}{\top}$$

(b) If $|\phi|_\vee = n > 0$, inspection of the rules provides us the following possible cases:

- $\phi =_\vee \dfrac{\dfrac{}{(A' \boxtimes B) \vee C}\,\phi'}{\boxed{\mathsf{r}\,\dfrac{(A' \boxtimes B) \vee C}{(A \boxtimes B) \vee C}}}$

  Since $|\phi'|_\vee =_\vee n - 1$, we apply induction hypothesis to ϕ'. There exist Q_1 and Q_2 and

$$\dfrac{\dfrac{Q_1 \;\boxtimes\; Q_2}{\dfrac{}{C}\,\psi}}{C}\,, \qquad \phi_1 \equiv \dfrac{\dfrac{}{\boxed{\mathsf{r}\,\dfrac{A'}{A}} \vee Q_1}\,\phi_1'}{\vee\, Q_1}\,, \qquad \dfrac{\dfrac{}{B \vee Q_2}\,\phi_2}{B \vee Q_2}$$

  such that $|\phi_1|_\vee + |\phi_2|_\vee = |\phi_1'|_\vee + |\phi_2|_\vee + 1 = |\phi|_\vee$.
  If ϕ is tame, then ψ, ϕ_1' and ϕ_2 are tame. Furthermore, since ϕ is tame, r is tame and therefore ϕ_1 is tame.

- $\phi =_\vee \dfrac{\overset{\phi'}{\Vdash}}{\mathsf{r}\dfrac{(A \boxtimes B') \vee C}{(A \boxtimes B) \vee C}}$

  We apply induction hypothesis to ϕ'. There exist Q_1 and Q_2 and

  $$\begin{array}{c} Q_1 \ \boxtimes \ Q_2 \\ \psi \Vdash \\ C \end{array}, \qquad \begin{array}{c} \phi_1 \Vdash \\ A \vee Q_1 \end{array}, \qquad \phi_2 \equiv \mathsf{r}\dfrac{\overset{\phi_2'}{\Vdash}}{\dfrac{B'}{B}} \vee Q_2$$

  such that $|\phi_1|_\vee + |\phi_2|_\vee = |\phi_1|_\vee + |\phi_2'|_\vee + 1 \leq |\phi'|_\vee + 1 = |\phi|_\vee$.
  If ϕ is tame, then ψ, ϕ_1 and ϕ_2' are tame. Furthermore, since ϕ is tame, r is tame and therefore ϕ_2 is tame.

- $\phi =_\vee \dfrac{\overset{\phi'}{\Vdash}}{\mathsf{r}\dfrac{(A \boxtimes B) \vee C'}{(A \boxtimes B) \vee C}}$

  We apply induction hypothesis to ϕ'. There exist Q_1, Q_2 and

  $$\psi \equiv \begin{array}{c} Q_1 \ \boxtimes \ Q_2 \\ \psi' \Vdash \\ \mathsf{r}\dfrac{C'}{C} \end{array}, \qquad \begin{array}{c} \phi_1 \Vdash \\ A \vee Q_1 \end{array}, \qquad \begin{array}{c} \phi_2 \Vdash \\ B \vee Q_2 \end{array}$$

  such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi'|_\vee < |\phi|_\vee$.
  If ϕ is tame, then ψ', ϕ_1 and ϕ_2 are tame. Furthermore, since ϕ is tame, r is tame and therefore ψ is tame.

- $\phi =_\vee \dfrac{\overset{\phi'}{\Vdash}}{\wedge_\downarrow \dfrac{(((A \boxtimes B) \vee C_1) \wedge (C_2 \vee C_3)) \vee C_4}{(A \boxtimes B) \vee C_2 \vee (C_1 \wedge C_3) \vee C_4}}$ with $C = C_2 \vee (C_1 \wedge C_3) \vee C_4$.

  We apply induction hypothesis to ϕ' since $|\phi'|_\vee < |\phi|_\vee$. There exist Q_1', Q_2' and

  $$\begin{array}{c} Q_1' \ \vee \ Q_2' \\ \psi' \Vdash \\ C_4 \end{array}, \qquad \begin{array}{c} \phi_1' \Vdash \\ (A \boxtimes B) \vee C_1 \vee Q_1' \end{array}, \qquad \begin{array}{c} \phi_2' \Vdash \\ C_2 \vee C_3 \vee Q_2' \end{array}$$

  such that $|\phi_1'|_\vee + |\phi_2'|_\vee \leq |\phi'|_\vee$.
  We apply induction hypothesis to ϕ_1'. There exist Q_1, Q_2 and

$$\psi \equiv \dfrac{\dfrac{\dfrac{Q_1 \boxtimes Q_2}{\psi''\Big\|} \quad \wedge \quad \dfrac{}{\phi_2'\Big\|\ C_3 \vee C_2 \vee Q_2'}}{C_1 \vee Q_1'}}{Q_1' \vee Q_2'}\wedge_{\downarrow}, \quad \dfrac{}{\phi_1\Big\|\ A \vee Q_1}, \quad \dfrac{}{\phi_2\Big\|\ B \vee Q_2}$$
$$= \dfrac{\dfrac{(C_1 \wedge C_3) \vee C_2 \vee \quad \dfrac{}{\psi'\Big\|\ C_4}}{C}}{C}=_{\vee}$$

such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi_1'|_\vee \leq |\phi|_\vee$.
If ϕ is tame, then ψ', ϕ_1' and ϕ_2' are tame. Therefore, ψ'', ϕ_1 and ϕ_2 are tame and thus ψ is tame.

- $\phi =_\vee \boxed{\mathsf{sak}_\downarrow \dfrac{\dfrac{}{\phi'\Big\|\ ((A \vee C_1) \boxtimes (B \vee C_2)) \vee C_3}}{(A \boxtimes B) \vee (C_1 \boxtimes C_2) \vee C_3}}$ with $C = (C_1 \boxtimes C_2) \vee C_3$

We apply induction hypothesis to ϕ'. There exist Q_1', Q_2' and

$$\dfrac{Q_1' \ \boxtimes \ Q_2'}{\psi'\Big\|\ C_3}, \quad \dfrac{}{\phi_1'\Big\|\ A \vee C_1 \vee Q_1'}, \quad \dfrac{}{\phi_2'\Big\|\ (A \boxtimes B) \vee C_2 \vee Q_2'}$$

such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi_1'|_\vee \leq |\phi|_\vee$.
We take $Q_1 = C_1 \vee Q_1'$ and $Q_2 = C_2 \vee Q_2'$, then we have

$$\psi \equiv \dfrac{\dfrac{\dfrac{Q_1 \boxtimes Q_2}{(C_1 \vee Q_1') \boxtimes (C_2 \vee Q_2')}\mathsf{sak}_\downarrow}{Q_1' \boxtimes Q_2'}}{(C_1 \boxtimes C_2) \vee \quad \dfrac{}{\psi'\Big\|\ C_3}}, \quad \dfrac{}{\phi_1\Big\|\ A \vee Q_1}, \quad \dfrac{}{\phi_2\Big\|\ B \vee Q_2}$$
$$= \dfrac{}{C}=_{\vee}$$

If ϕ is tame, then ψ is tame and ϕ_1 and ϕ_2 are equalities.

- $\phi =_\vee \boxed{=_\vee \dfrac{\dfrac{}{\phi'\Big\|\ (\top \wedge ((A \boxtimes B) \vee C_1)) \vee C_2}}{(A \boxtimes B) \vee C_1 \vee C_2}}$ with $C = C_1 \vee C_2$

We apply induction hypothesis to ϕ'. There exist Q_1', Q_2' and

$$\dfrac{Q_1' \ \vee \ Q_2'}{\psi'\Big\|\ C_2}, \quad \dfrac{}{\phi_1'\Big\|\ 1 \vee Q_1'}, \quad \dfrac{}{\phi_2'\Big\|\ (A \boxtimes B) \vee C_1 \vee Q_2'}$$

Then we apply induction hypothesis to ϕ_2'. There exist Q_1, Q_2 and

$$\psi \equiv \dfrac{\dfrac{\dfrac{}{(1 \vee Q_1')}\,{\scriptstyle \phi_1'}\!\Big\|}{(1 \wedge C_1) \vee Q_1'}\,\wedge_{\downarrow} \dfrac{\dfrac{}{(C_1 \vee Q_2')}\,{\scriptstyle \psi''}\!\Big\|}{Q_1' \vee Q_2'}\,{\scriptstyle \psi'}\!\Big\|}{C}\,\Big\|\!\!\Big\|\,\dfrac{Q_1 \boxtimes Q_2}{C_2}, \qquad \dfrac{}{A \vee Q_1}\,{\scriptstyle \phi_1}\!\Big\|, \qquad \dfrac{}{B \vee Q_2}\,{\scriptstyle \phi_2}\!\Big\|$$

such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi_2'|_\vee \leq |\phi'|_\vee \leq |\phi|_\vee$.
If ϕ is tame, then ψ', ϕ_1', ϕ_2' and ψ'' are tame. Therefore, ψ is tame. Furthermore, by induction hypothesis, ϕ_1 and ϕ_2 are equalities.

- $\phi =_\vee \boxed{\dfrac{\dfrac{}{(((A \boxtimes B) \vee C_1) \wedge \top) \vee C_2}\,{\scriptstyle \phi'}\!\Big\|}{(A \boxtimes B) \vee C_1 \vee C_2}\,=_\vee}$ with $C = C_1 \vee C_2$

We apply induction hypothesis to ϕ' as $|\phi'|_\vee \leq |\phi'|_\vee$. There exist Q_1', Q_2' and

$$\dfrac{Q_1' \;\vee\; Q_2'}{C_2}\,{\scriptstyle \psi'}\!\Big\|, \qquad \dfrac{}{(A \boxtimes B) \vee C_1 \vee Q_1'}\,{\scriptstyle \phi_1'}\!\Big\|, \qquad \dfrac{}{1 \vee Q_2'}\,{\scriptstyle \phi_2'}\!\Big\|$$

with $|\phi_1'|_\vee + |\phi_2'|_\vee \leq |\phi'|_\vee$.
Then we apply induction hypothesis to ϕ_1'. We have Q_1, Q_2 and

$$\psi \equiv \dfrac{\dfrac{\dfrac{}{(C_1 \vee Q_1')}\,{\scriptstyle \psi''}\!\Big\|}{(C_1 \wedge 1) \vee Q_1'}\,\wedge_{\downarrow} \dfrac{\dfrac{}{(1 \vee Q_2')}\,{\scriptstyle \phi_2'}\!\Big\|}{Q_1' \vee Q_2'}\,{\scriptstyle \psi'}\!\Big\|}{C}\,\Big\|\!\!\Big\|\,\dfrac{Q_1 \boxtimes Q_2}{C_2}, \qquad \dfrac{}{A \vee Q_1}\,{\scriptstyle \phi_1}\!\Big\|, \qquad \dfrac{}{B \vee Q_2}\,{\scriptstyle \phi_2}\!\Big\|$$

such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi_1'|_\vee \leq |\phi'|_\vee \leq |\phi|_\vee$.
If ϕ is tame, then ψ', ϕ_1', ϕ_2' and ψ'' are tame. Therefore, ψ is tame. Furthermore, by induction hypothesis, ϕ_1 and ϕ_2 are equalities.

- $\phi =_\vee \boxed{\dfrac{\dfrac{}{\top \vee C}\,{\scriptstyle \phi'}\!\Big\|}{(\top \boxtimes \top) \vee C}\,=_\vee}$ with $A =_\vee B =_\vee \top$

We take:

$$\phi_1 \equiv \dfrac{\dfrac{\top}{\boxed{\dfrac{\top}{A}\,=_\vee} \vee \bot}}{}\,=_\vee, \qquad \phi_2 \equiv \dfrac{\dfrac{\top}{\boxed{\dfrac{\top}{B}\,=_\vee} \vee \bot}}{}\,=_\vee \qquad \text{and}$$

$$\psi \equiv \wedge_{\downarrow} \dfrac{\boxed{=\dfrac{\bot \boxtimes \bot}{\bot}} \vee \bot \wedge \overset{\phi'}{\Vert}(\top \vee C)}{\boxed{=\dfrac{\bot \wedge \top}{\bot}} \vee \bot \vee C}$$

with $|\phi_1|_\vee = |\phi_2|_\vee = 0$.
If ϕ is tame, ψ is tame. Furthermore, ϕ_1 and ϕ_2 are equalities.

- $\phi =_\vee \boxed{=_\vee \dfrac{\overset{\phi'}{\Vert}\bot \vee C}{(\bot \boxtimes \bot) \vee C}}$ with $A =_\vee B =_\vee \bot$

  We take:

$$\phi_1 \equiv {}^{=_\vee}\dfrac{\top}{\boxed{=_\vee \dfrac{\bot}{A}} \vee \top}, \qquad \phi_2 \equiv {}^{=_\vee}\dfrac{\top}{\boxed{=_\vee \dfrac{\bot}{B}} \vee \top} \qquad \text{and}$$

$$\psi \equiv \wedge_{\downarrow} \dfrac{\boxed{=\dfrac{\top \boxtimes \top}{\top}} \vee \bot \wedge \overset{\phi'}{\Vert}(\top \vee C)}{\boxed{=\dfrac{\top \wedge \bot}{\bot}} \vee \bot \vee C}$$

  with $|\phi_1|_\vee = |\phi_2|_\vee = 0$.
  If ϕ is tame, ψ is tame. Furthermore, ϕ_1 and ϕ_2 are equalities.

2. Given a proof ϕ of $(A \wedge B) \vee C$ in $\mathsf{SAK^KS_{sp}}$. We proceed by induction on $|\phi|_\vee$.

   (a) If $|\phi|_\vee =_\vee 0$, then $(A \wedge B) \vee C =_\vee \top$. Then, either:

   - $A =_\vee B =_\vee \top, C =_\vee \bot$ and we take

$$\psi \equiv \dfrac{=\dfrac{\bot \vee \bot}{\bot}}{=_\vee \overline{\overline{C}}}, \quad \phi_1 \equiv {}^{=_\vee}\dfrac{\top}{\boxed{=_\vee \dfrac{\top}{A}} \vee \bot}, \quad \phi_2 \equiv {}^{=_\vee}\dfrac{\top}{\boxed{=_\vee \dfrac{\top}{B}} \vee \bot}$$

   - $A =_\vee \bot, B =_\vee C =_\vee \top$ and we take $Q_1 = \top$ and $Q_2 = \bot$:

$$\psi \equiv \dfrac{=\dfrac{\top \vee \bot}{\top}}{=_\vee \overline{\overline{C}}}, \quad \phi_1 \equiv {}^{=_\vee}\dfrac{\top}{\boxed{=_\vee \dfrac{\bot}{A}} \vee \top}, \quad \phi_2 \equiv {}^{=_\vee}\dfrac{\top}{\boxed{=_\vee \dfrac{\top}{B}} \vee \bot}$$

   - $B =_\vee \bot, A =_\vee C =_\vee \top$ and we take $Q_1 = \bot$ and $Q_2 = \top$:

$$\psi \equiv \dfrac{=\dfrac{\bot \vee \top}{\top}}{=_\vee \overline{\overline{C}}}, \quad \phi_1 \equiv {}^{=_\vee}\dfrac{\top}{\boxed{=_\vee \dfrac{\top}{A}} \vee \bot}, \quad \phi_2 \equiv {}^{=_\vee}\dfrac{\top}{\boxed{=_\vee \dfrac{\bot}{B}} \vee \top}$$

   (b) If $|\phi|_\vee = n > 0$, inspection of the rules provides us the following possible cases:

- $\phi =_\vee \dfrac{\dfrac{\dfrac{}{(A' \wedge B) \vee C}\,\phi'}{(A \wedge B) \vee C}\,\mathsf{r}}$

  Since $|\phi'|_\vee =_\vee n - 1$, we apply induction hypothesis to ϕ'. There exist Q_1 and Q_2 and

  $$\begin{array}{c} Q_1 \ \vee \ Q_2 \\ \psi \| \\ C \end{array}, \qquad \phi_1 \equiv \dfrac{\dfrac{}{A'}\,\phi_1'}{A}\,\mathsf{r} \vee Q_1, \qquad \begin{array}{c} \\ \phi_2 \| \\ B \vee Q_2 \end{array}$$

  such that $|\phi_1|_\vee + |\phi_2|_\vee = |\phi_1'|_\vee + |\phi_2|_\vee + 1 = |\phi|_\vee$.
  If ϕ is tame, then ψ, ϕ_1' and ϕ_2 are tame. Furthermore, since ϕ is tame, then r is tame, and therefore ϕ_1 is tame.

- $\phi =_\vee \dfrac{\dfrac{\dfrac{}{(A \wedge B') \vee C}\,\phi'}{(A \wedge B) \vee C}\,\mathsf{r}}$

  Since $|\phi'|_\vee =_\vee n - 1$, we apply induction hypothesis to ϕ'. There exist Q_1 and Q_2 and

  $$\begin{array}{c} Q_1 \ \vee \ Q_2 \\ \psi \| \\ C \end{array}, \qquad \begin{array}{c} \phi_1 \| \\ A \vee Q_1 \end{array}, \qquad \phi_2 \equiv \dfrac{\dfrac{}{B'}\,\phi_2'}{B}\,\mathsf{r} \vee Q_2$$

  such that $|\phi_1|_\vee + |\phi_2|_\vee = |\phi_1|_\vee + |\phi_2'|_\vee + 1 \leq |\phi'|_\vee + 1 = |\phi|_\vee$.
  If ϕ is tame, then ψ, ϕ_1 and ϕ_2' are tame. Furthermore, since ϕ is tame, then r is tame, and therefore ϕ_2 is tame.

- $\phi =_\vee \dfrac{\dfrac{\dfrac{}{(A \wedge B) \vee C'}\,\phi'}{(A \wedge B) \vee C}\,\mathsf{r}}$

  We apply induction hypothesis to ϕ'. There exist Q_1, Q_2 and

  $$\psi \equiv \begin{array}{c} Q_1 \ \vee \ Q_2 \\ \dfrac{\dfrac{}{C'}\,\psi'}{C}\,\mathsf{r} \end{array}, \qquad \begin{array}{c} \phi_1 \| \\ A \vee Q_1 \end{array}, \qquad \begin{array}{c} \phi_2 \| \\ B \vee Q_2 \end{array}$$

  such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi'|_\vee \leq |\phi|_\vee$.
  If ϕ is tame, then ψ', ϕ_1 and ϕ_2 are tame. Furthermore, since ϕ is tame, then r is tame. Therefore ψ is tame.

- $\phi =_\vee \boxed{\wedge_\downarrow \dfrac{\overset{\phi'}{\Vert}\ ((A \vee C_1) \wedge (B \vee C_2)) \vee C_3}{(A \wedge B) \vee C_1 \vee C_2 \vee C_3}}$ with $C = C_1 \vee C_2 \vee C_3$

  We apply induction hypothesis to ϕ'. There exist Q'_1, Q'_2 and

  $$\begin{array}{c} Q'_1 \ \vee \ Q'_2 \\ \psi' \Vert \\ C_3 \end{array}, \qquad \overset{\phi_1}{\Vert}\ A \vee C_1 \vee Q'_1, \qquad \overset{\phi_2}{\Vert}\ B \vee C_2 \vee Q'_2$$

  such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi'|_\vee \leq |\phi|_\vee$.
  We take $Q_1 = C_1 \vee Q'_1$ and $Q_2 = C_2 \vee Q'_2$ and we have

  $$\psi \equiv \begin{array}{l} \\ =_\vee \dfrac{\dfrac{Q_1 \vee Q_2}{C_1 \vee Q'_1 \vee C_2 \vee Q'_2}}{Q'_1 \vee Q'_2} \\ C_1 \vee C_2 \vee \ \psi' \Vert \\ =_\vee \dfrac{}{C} \end{array} \begin{array}{c} \\ \\ \\ C_3 \\ \hline\hline \end{array}$$

  If ϕ is tame, then ϕ' is tame and by induction hypothesis ϕ_1, ϕ_2 and ψ' are tame.
  Since ϕ and ψ' are tame and C_1, Q'_1, C_2, Q'_2 are interpretable, then ψ is tame.

- $\phi =_\vee \boxed{\wedge_\downarrow \dfrac{\overset{\phi'}{\Vert}\ (((A \wedge B) \vee C_1) \wedge (C_2 \vee C_3)) \vee C_4}{(A \wedge B) \vee C_2 \vee (C_1 \wedge C_3) \vee C_4}}$

  with $C = C_2 \vee (C_1 \wedge C_3) \vee C_4$.

  We apply induction hypothesis to ϕ'. There exist Q'_1, Q'_2 and

  $$\begin{array}{c} Q'_1 \ \vee \ Q'_2 \\ \psi' \Vert \\ C_4 \end{array}, \qquad \overset{\phi'_1}{\Vert}\ (A \wedge B) \vee C_1 \vee Q'_1, \qquad \overset{\phi'_2}{\Vert}\ C_2 \vee C_3 \vee Q'_2$$

  such that $|\phi'_1|_\vee + |\phi'_2|_\vee \leq |\phi'|_\vee$.
  We apply induction hypothesis to ϕ'_1. There exist Q_1, Q_2 and

  $$\psi \equiv \begin{array}{l} \\ \wedge_\downarrow \dfrac{\dfrac{Q_1 \vee Q_2}{\psi'' \Vert \quad C_1 \vee Q'_1} \wedge \overset{\phi'_2}{\Vert}\ C_3 \vee C_2 \vee Q'_2}{Q'_1 \vee Q'_2}, \\ (C_1 \wedge C_3) \vee C_2 \vee \ \psi' \Vert \\ =_\vee \dfrac{}{C} \end{array} \begin{array}{c} \\ \\ \\ C_4 \\ \hline\hline \end{array} \qquad \begin{array}{l} \overset{\phi_1}{\Vert}\ A \vee Q_1, \\ \\ \\ \end{array} \qquad \begin{array}{l} \overset{\phi_2}{\Vert}\ B \vee Q_2 \\ \\ \\ \end{array}$$

such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi'_1|_\vee \leq |\phi|_\vee$.
If ϕ is tame, then ϕ'_1 is tame and ϕ_1, ϕ_2 and ψ'' are tame. ψ' and ϕ'_2 are tame as well, and since the interpretation is preservable, C_1, C_2, C_3 are interpretable. Therefore ψ is tame.

- $\phi =_\vee \boxed{\begin{array}{c} {\scriptstyle \phi'}\| \\ =_\vee \dfrac{(A_1 \wedge (A_2 \wedge B)) \vee C}{((A_1 \wedge A_2) \wedge B) \vee C} \end{array}}$

We apply induction hypothesis to ϕ'. There exist Q'_1, Q'_2 and

$$\begin{array}{c} Q'_1 \ \vee \ Q'_2 \\ {\scriptstyle \psi'}\| \\ C \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi'_1}\| \\ A_1 \vee Q'_1 \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi'_2}\| \\ (A_2 \wedge B) \vee Q'_2 \end{array}$$

with $|\phi'_1|_\vee + |\phi'_2|_\vee \leq |\phi'|_\vee$.
We apply induction hypothesis to ϕ'_2, then there exist Q''_1, Q_2 and

$$\begin{array}{c} Q''_1 \ \vee \ Q_2 \\ {\scriptstyle \psi''}\| \\ Q'_2 \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi''_1}\| \\ A_2 \vee Q''_1 \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi'_2}\| \\ B \vee Q_2 \end{array}$$

such that $|\phi''_1|_\vee + |\phi_2|_\vee \leq |\phi'_2|_\vee$.
We take $Q_1 \equiv Q'_1 \vee Q''_1$ and

$$\psi \equiv \begin{array}{c} \\ =_\vee \dfrac{Q_1 \vee Q_2}{Q'_1 \vee \boxed{\begin{array}{c} Q''_1 \vee Q_2 \\ {\scriptstyle \psi''}\| \\ Q'_2 \end{array}}} \\ {\scriptstyle \psi'}\| \\ C \end{array}, \qquad \phi_1 \equiv \begin{array}{c} {\scriptstyle \phi'_1}\| \qquad {\scriptstyle \phi''_1}\| \\ \wedge_\downarrow \dfrac{(A_1 \vee Q'_1) \ \wedge \ (A_2 \vee Q''_1)}{(A_1 \wedge A_2) \vee Q'_1 \vee Q''_1} \end{array}$$

and we have $|\phi_1|_\vee + |\phi_2|_\vee \equiv |\phi'_1|_\vee + |\phi''_1|_\vee + 1 + |\phi_2|_\vee \leq |\phi'_1|_\vee + |\phi'_2|_\vee + 1 \leq |\phi'|_\vee + 1 = |\phi|_\vee$.
If ϕ is tame, then ϕ'_1, ϕ''_1, ψ' and ψ'' are tame and so Q_2, Q'_1, Q''_1 are interpretable. Therefore ϕ_1, ϕ_2 and ψ are tame.

- $\phi =_\vee \boxed{\begin{array}{c} {\scriptstyle \phi'}\| \\ =_\vee \dfrac{((A \wedge B_1) \wedge B_2) \vee C}{(A \wedge (B_1 \wedge B_2)) \vee C} \end{array}}$ with $B = B_1 \wedge B_2$

We apply induction hypothesis to ϕ'. There exist Q'_1, Q'_2 and

$$\begin{array}{c} Q'_1 \ \vee \ Q'_2 \\ {\scriptstyle \psi'}\| \\ C \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi'_1}\| \\ (A \wedge B_1) \vee Q'_1 \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi'_2}\| \\ B_2 \vee Q'_2 \end{array}$$

with $|\phi'_1|_\vee + |\phi'_2|_\vee \leq |\phi'|_\vee$.
Then we apply induction hypothesis to ϕ'_1. There exist Q_1, Q''_2 and we have

$$\begin{array}{c} Q_1 \ \vee \ Q_2'' \\ {\scriptstyle \psi''}\Big\| \\ Q_1' \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi_1}\Big\| \\ A \vee Q_1 \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi_2''}\Big\| \\ B_1 \vee Q_2'' \end{array}$$

We take $Q_2 \equiv Q_2'' \vee Q_2'$ and we get

$$\psi \equiv \begin{array}{c} =_\vee \dfrac{Q_1 \vee Q_2}{\boxed{\begin{array}{c} Q_1 \vee Q_2'' \\ {\scriptstyle \psi''}\Big\| \\ Q_1' \end{array}} \vee Q_2'} \\ {\scriptstyle \psi'}\Big\| \\ C \end{array}, \qquad \phi_2 \equiv \begin{array}{c} \wedge_\downarrow \dfrac{\begin{array}{cc} {\scriptstyle \phi_2''}\Big\| & {\scriptstyle \phi_2'}\Big\| \\ (B_1 \vee Q_2'') & \wedge \ (B_2 \vee Q_2') \end{array}}{(B_1 \wedge B_2) \vee Q_2'' \vee Q_2'} \end{array}$$

and we have $|\phi_1|_\vee + |\phi_2|_\vee \equiv |\phi_1|_\vee + |\phi_2'|_\vee + |\phi_2''|_\vee + 1 \leq |\phi_1'|_\vee + |\phi_2'|_\vee + 1 \leq |\phi'|_\vee + 1 = |\phi|_\vee$.
If ϕ is tame, then ϕ_2', ϕ_2'', ψ' and ψ'' are tame and so Q_1, Q_2', Q_2'' are interpretable. Therefore ϕ_1, ϕ_2 and ψ are tame.

- $\phi =_\vee \begin{array}{c} {\scriptstyle \phi'}\Big\| \\ \boxed{=_\vee \dfrac{(A \wedge B) \vee C}{(B \wedge A) \vee C}} \end{array}$

  Since $|\phi'|_\vee =_\vee n - 1$, we apply induction hypothesis to ϕ'. There exist Q_1 and Q_2 and

  $$\begin{array}{c} Q_2 \ \vee \ Q_1 \\ {\scriptstyle \psi'}\Big\| \\ C \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi_1}\Big\| \\ A \vee Q_2 \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi_2}\Big\| \\ B \vee Q_1 \end{array}$$

  Then we have $\psi \equiv \begin{array}{c} =_\vee \dfrac{Q_1 \vee Q_2}{Q_2 \vee Q_1} \\ {\scriptstyle \psi'}\Big\| \\ C \end{array}$.

  If ϕ is tame, then ϕ' is tame and by induction hypothesis ϕ_1, ϕ_2 and ψ' are tame. Then Q_1 and Q_2 are interpretable and hence ψ is tame as well.

- $\phi =_\vee \begin{array}{c} {\scriptstyle \phi'}\Big\| \\ \boxed{=_\vee \dfrac{(((A \wedge B) \vee C_1) \wedge \top) \vee C_2}{(A \wedge B) \vee C_1 \vee C_2}} \end{array}$ with $C = C_1 \vee C_2$

  We apply induction hypothesis to ϕ'. There exist Q_1', Q_2' and

  $$\begin{array}{c} Q_1' \ \vee \ Q_2' \\ {\scriptstyle \psi'}\Big\| \\ C_2 \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi_1'}\Big\| \\ (A \wedge B) \vee C_1 \vee Q_1' \end{array}, \qquad \begin{array}{c} {\scriptstyle \phi_2'}\Big\| \\ 1 \vee Q_2' \end{array}$$

  Then we apply induction hypothesis to ϕ_1'. We have Q_1, Q_2 and

$$\psi \equiv \dfrac{\dfrac{\dfrac{Q_1 \vee Q_2}{\psi''\Big\|} \wedge \dfrac{}{\phi_2'\Big\|}}{\dfrac{(C_1 \vee Q_1')}{Q_1' \vee Q_2'}\wedge_{\downarrow}} \quad \dfrac{(1 \vee Q_2')}{}}{\dfrac{(C_1 \wedge 1) \vee \dfrac{}{\psi'\Big\|}}{C}} \qquad \dfrac{}{\phi_1\Big\|}{A \vee Q_1}, \qquad \dfrac{}{\phi_2\Big\|}{B \vee Q_2}$$

such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi_1'|_\vee \leq |\phi'|_\vee \leq |\phi|_\vee$.
If ϕ is tame, then so are ψ', ϕ_1' and ϕ_2'. Therefore, ψ'', ϕ_1 and ϕ_2 are tame, and so is ψ.

- $\phi =_\vee \boxed{\dfrac{\dfrac{}{\phi'\Big\|}}{(\top \wedge ((A \wedge B) \vee C_1)) \vee C_2}=_\vee}$ with $C = C_1 \vee C_2$

We apply induction hypothesis to ϕ'. There exist Q_1', Q_2' and

$$\dfrac{Q_1' \vee Q_2'}{\psi'\Big\|}{C_2}, \qquad \dfrac{}{\phi_1'\Big\|}{1 \vee Q_1'}, \qquad \dfrac{}{\phi_2'\Big\|}{(A \wedge B) \vee C_1 \vee Q_2'}$$

Then we apply induction hypothesis to ϕ_2'. There exist Q_1, Q_2 and

$$\psi \equiv \dfrac{\dfrac{\dfrac{}{\phi_1'\Big\|}{(1 \vee Q_1')} \wedge \dfrac{Q_1 \vee Q_2}{\psi''\Big\|}}{\dfrac{(C_1 \vee Q_2')}{Q_1' \vee Q_2'}\wedge_{\downarrow}}}{\dfrac{(1 \wedge C_1) \vee \dfrac{}{\psi'\Big\|}}{C}} \qquad \dfrac{}{\phi_1\Big\|}{A \vee Q_1}, \qquad \dfrac{}{\phi_2\Big\|}{B \vee Q_2}$$

such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi_2'|_\vee \leq |\phi'|_\vee \leq |\phi|_\vee$.
If ϕ is tame, then so are ψ', ϕ_1' and ϕ_2'. Therefore, ψ'', ϕ_1 and ϕ_2 are tame, and so is ψ.

- $\phi =_\vee \boxed{\dfrac{\dfrac{}{\phi'\Big\|}}{A \vee C}=_\vee}$ with $B =_\vee 1$

We apply induction hypothesis in ϕ' and we have the following derivations:

$$\psi \equiv \dfrac{C \vee \bot}{C}=_\vee, \qquad \phi_1 \equiv \dfrac{}{\phi_1'\Big\|}{A \vee C}, \qquad \phi_2 \equiv \dfrac{1}{\boxed{\dfrac{1}{B}=_\vee} \vee \bot}=_\vee$$

such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi|_\vee$.

If ϕ is tame, then C is interpretable and ϕ' is tame and thus ψ ϕ_1 and ϕ_2 are tame.

- $\phi =_\vee \boxed{=_\vee \dfrac{\overset{\phi'}{\Vert}\ B \vee C}{(1 \wedge B) \vee C}}$ with $A =_\vee 1$

  Then we take:

$$\psi \equiv =_\vee \frac{\bot \vee C}{C}, \qquad \phi_1 \equiv \overset{=_\vee}{\dfrac{\dfrac{1}{\boxed{=_\vee \dfrac{1}{A}} \vee \bot}}}, \qquad \phi_2 \equiv \overset{\phi_2'}{\Vert}\ B \vee C$$

  and we have $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi|_\vee$.
  If ϕ is tame, then C is interpretable and ϕ' is tame and thus ψ ϕ_1 and ϕ_2 are tame.

3. Given a proof ϕ of $(AaB) \vee C$ in $\mathsf{SAK^K S_{sp}}$. We proceed by induction on $|\phi|_\vee$.

   (a) If $|\phi|_\vee =_\vee 0$, then either:

   i) $A =_\vee B =_\vee \top, C =_\vee \bot$ and we take

$$\psi \equiv \overset{=}{\underset{=_\vee}{\dfrac{\bot a \bot}{\dfrac{\bot}{C}}}}, \quad \phi_1 \equiv \overset{=_\vee}{\dfrac{\dfrac{\top}{\boxed{=_\vee \dfrac{\top}{A}} \vee \ \bot}}}, \quad \phi_2 \equiv \overset{=_\vee}{\dfrac{\dfrac{\top}{\boxed{=_\vee \dfrac{\top}{B}} \vee \bot}}}$$

   with $|\phi_1|_\vee = |\phi_2|_\vee = 0$.

   ii) $A =_\vee B =_\vee \bot, C =_\vee \top$ and we take

$$\psi \equiv \overset{=}{\underset{=_\vee}{\dfrac{\top a \top}{\dfrac{\top}{C}}}}, \quad \phi_1 \equiv \overset{=_\vee}{\dfrac{\dfrac{\top}{\boxed{=_\vee \dfrac{\bot}{A}} \vee \ \top}}}, \quad \phi_2 \equiv \overset{=_\vee}{\dfrac{\dfrac{\top}{\boxed{=_\vee \dfrac{\bot}{B}} \vee \top}}}$$

   with $|\phi_1|_\vee = |\phi_2|_\vee = 0$.

   (b) If $|\phi|_\vee = n > 0$, inspection of the rules provides us the following possible cases:

- $\phi =_\vee \boxed{\mathsf{r}\dfrac{\overset{\phi'}{\Vert}\ (A'aB) \vee C}{(AaB) \vee C}}$

  Since $|\phi'|_\vee =_\vee n - 1$, we apply induction hypothesis to ϕ'. There exist Q_1, Q_2 and

$$\overset{\psi}{\underset{C}{\dfrac{Q_1\ a\ Q_2}{\Vert}}}, \qquad \phi_1 \equiv \boxed{\mathsf{r}\dfrac{\overset{\phi_1'}{\Vert}\ A'}{A}} \vee Q_1, \qquad \overset{\phi_2}{\Vert}\ B \vee Q_2$$

  such that $|\phi_1|_\vee + |\phi_2|_\vee = |\phi_1'|_\vee + |\phi_2|_\vee + 1 = |\phi|_\vee$.
  If ϕ is tame, ψ is tame and ϕ_1' and ϕ_2 are equalities. r is also an equality and therefore ϕ_1 is an equality.

- $\phi =_\vee \dfrac{\overset{\phi'}{\Vert}}{\boxed{\mathsf{r}\dfrac{(AaB') \vee C}{(AaB) \vee C}}}$

  We apply induction hypothesis to ϕ'. There exist Q_1, Q_2 and

  $$\dfrac{Q_1\ a\ Q_2}{\overset{\psi}{\Vert}\ C}\ , \qquad \dfrac{\overset{\phi_1}{\Vert}}{A \vee Q_1}, \qquad \phi_2 \equiv \boxed{\mathsf{r}\dfrac{\overset{\phi_2'}{\Vert}\ B'}{B}} \vee Q_2$$

  such that $|\phi_1|_\vee + |\phi_2|_\vee = |\phi_1|_\vee + |\phi_2'|_\vee + 1 \leq |\phi'|_\vee + 1 = |\phi|_\vee$.
  If ϕ is tame, ψ is tame and ϕ_1 and ϕ_2' are equalities. r is also an equality and therefore ϕ_2 is an equality.

- $\phi =_\vee \dfrac{\overset{\phi'}{\Vert}}{\boxed{\mathsf{r}\dfrac{(AaB) \vee C'}{(AaB) \vee C}}}$

  We apply induction hypothesis to ϕ'. There exist Q_1, Q_2 and

  $$\dfrac{Q_1\ a\ Q_2}{\overset{\psi}{\Vert}\ \boxed{\mathsf{r}\dfrac{C'}{C}}}\ , \qquad \dfrac{\overset{\phi_1}{\Vert}}{A \vee Q_1}, \qquad \dfrac{\overset{\phi_2}{\Vert}}{B \vee Q_2}$$

  such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi'|_\vee < |\phi|_\vee$.
  If ϕ is tame so are ψ' and r, and therefor is ψ. ϕ_1 and ϕ_2 are equalities.

- $\phi =_\vee \dfrac{\overset{\phi'}{\Vert}}{\boxed{\mathsf{sai}_\downarrow \dfrac{((A \vee C_1)a(B \vee C_2)) \vee C_3}{(AaB) \vee (C_1aC_2) \vee C_3}}}$ with $C =_\vee (C_1aC_2) \vee C_3$

  We apply induction hypothesis to ϕ' as $|\phi'|_\vee < |\phi|_\vee$. There are derivations Q_1' and Q_2' such that

  $$\dfrac{Q_1'\ a\ Q_2'}{\overset{\psi}{\Vert}\ C_3}\ , \qquad \dfrac{\overset{\phi_1}{\Vert}}{A \vee C_1 \vee Q_1'}, \qquad \dfrac{\overset{\phi_2}{\Vert}}{B \vee C_2 \vee Q_2'}$$

  with $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi'|_\vee < |\phi|_\vee$.
  We take $Q_1 = C_1 \vee Q_1'$, $Q_2 = C_2 \vee Q_2'$ and

  $$\psi \equiv \mathsf{sai}_\downarrow \dfrac{\dfrac{Q_1aQ_2}{\dfrac{\overline{\overline{(C_1 \vee Q_1')a(C_2 \vee Q_2')}}}{Q_1'aQ_2'}}}{(C_1aC_2) \vee \overset{\psi'}{\Vert}\ C_3}$$

If ϕ is tame, then ψ' is tame and ϕ_1 and ϕ_2 are equalities. Then $C_1 \vee Q_1' = \top$ or $C_1 \vee Q_1' = \bot$ and $C_2 \vee Q_2' = \top$ or $C_2 \vee Q_2' = \bot$. Therefore, $(C_1 \vee Q_1')a(C_2 \vee Q_2')$ and $C_1 a C_2$ are interpretable and ψ is tame.

- $\phi =_\vee \boxed{\wedge_\downarrow \dfrac{\overset{\phi'}{\Vert}\ (((AaB) \vee C_1) \wedge (C_2 \vee C_3)) \vee C_4}{(AaB) \vee C_2 \vee (C_1 \wedge C_3) \vee C_4}}$ with $C = C_2 \vee (C_1 \wedge C_3) \vee C_4$.

We apply induction hypothesis to ϕ'. There exist Q_1', Q_2' and

$$\begin{matrix} Q_1' \ \vee \ Q_2' \\ {\scriptstyle \psi'}\Vert \\ C_4 \end{matrix}, \qquad \begin{matrix} {\scriptstyle \phi_1'}\Vert \\ (AaB) \vee C_1 \vee Q_1' \end{matrix}, \qquad \begin{matrix} {\scriptstyle \phi_2'}\Vert \\ C_2 \vee C_3 \vee Q_2' \end{matrix}$$

such that $|\phi_1'|_\vee + |\phi_2'|_\vee \leq |\phi'|_\vee$.
We apply induction hypothesis to ϕ_1' as $|\phi_1'|_\vee \leq |\phi'|_\vee < |\phi|_\vee$. There exist Q_1, Q_2 and

$$\psi \equiv \begin{array}{l} \wedge_\downarrow \dfrac{\begin{matrix} Q_1 a Q_2 \\ {\scriptstyle \psi''}\Vert \\ C_1 \vee Q_1' \end{matrix} \ \wedge \ \begin{matrix} {\scriptstyle \phi_2'}\Vert \\ C_3 \vee C_2 \vee Q_2' \end{matrix}}{Q_1' \vee Q_2'}, \\ =_\vee \dfrac{(C_1 \wedge C_3) \vee C_2 \vee \ {\scriptstyle \psi'}\Vert \ C_4}{C} \end{array} \qquad \begin{matrix} {\scriptstyle \phi_1}\Vert \\ A \vee Q_1 \end{matrix}, \qquad \begin{matrix} {\scriptstyle \phi_2}\Vert \\ B \vee Q_2 \end{matrix}$$

with $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi_1'|_\vee \leq |\phi'|_\vee \leq |\phi|_\vee$.
If ϕ is tame, then ψ', ϕ_1', ϕ_2' and ψ'' are tame. Therefore, ψ is tame. Furthermore, by induction hypothesis, ϕ_1 and ϕ_2 are equalities.

- $\phi =_\vee \boxed{=_\vee \dfrac{\overset{\phi'}{\Vert}\ (((AaB) \vee C_1) \wedge \top) \vee C_2}{(AaB) \vee C_1 \vee C_2}}$ with $C = C_1 \vee C_2$

We apply induction hypothesis to ϕ'. There exist Q_1', Q_2' and

$$\begin{matrix} Q_1' \ \vee \ Q_2' \\ {\scriptstyle \psi'}\Vert \\ C_2 \end{matrix}, \qquad \begin{matrix} {\scriptstyle \phi_1'}\Vert \\ (AaB) \vee C_1 \vee Q_1' \end{matrix}, \qquad \begin{matrix} {\scriptstyle \phi_2'}\Vert \\ 1 \vee Q_2' \end{matrix}$$

such that $|\phi_1'|_\vee + |\phi_2'|_\vee \leq |\phi'|_\vee$.
We apply induction hypothesis to ϕ_1'. Then there exist Q_1, Q_2 and

$$\psi \equiv \wedge_{\downarrow} \dfrac{\begin{array}{c} Q_1 a Q_2 \\ {\scriptstyle\psi''}\Vert \\ (C_1 \vee Q_1') \end{array} \wedge \begin{array}{c} {\scriptstyle\phi_2'}\Vert \\ (1 \vee Q_2') \end{array}}{\begin{array}{c} Q_1' \vee Q_2' \\ (C_1 \wedge 1) \vee \ {\scriptstyle\psi'}\Vert \\ C_2 \end{array}}, \qquad \begin{array}{c} {\scriptstyle\phi_1}\Vert \\ A \vee Q_1 \end{array}, \qquad \begin{array}{c} {\scriptstyle\phi_2}\Vert \\ B \vee Q_2 \end{array}$$

such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi_1'|_\vee \leq |\phi'|_\vee \leq |\phi|_\vee$.
If ϕ is tame, then ψ', ϕ_1', ϕ_2' and ψ'' are tame. Therefore, ψ is tame. Furthermore, by induction hypothesis, ϕ_1 and ϕ_2 are equalities.

- $\phi =_\vee \boxed{\begin{array}{c} {\scriptstyle\phi'}\Vert \\ =_\vee \dfrac{(\top \wedge ((AaB) \vee C_1)) \vee C_2}{(AaB) \vee C_1 \vee C_2} \end{array}}$ with $C = C_1 \vee C_2$

We apply induction hypothesis to ϕ'. There exist Q_1', Q_2' and

$$\begin{array}{c} Q_1' \ \vee \ Q_2' \\ {\scriptstyle\psi'}\Vert \\ C_2 \end{array}, \qquad \begin{array}{c} {\scriptstyle\phi_1'}\Vert \\ 1 \vee Q_1' \end{array}, \qquad \begin{array}{c} {\scriptstyle\phi_2'}\Vert \\ (AaB) \vee C_1 \vee Q_2' \end{array}$$

such that $|\phi_1'|_\vee + |\phi_2'|_\vee \leq |\phi'|_\vee$.
Then we apply induction hypothesis to ϕ_2'. There exist Q_1, Q_2 and

$$\psi \equiv \wedge_{\downarrow} \dfrac{\begin{array}{c} {\scriptstyle\phi_1'}\Vert \\ (1 \vee Q_1') \end{array} \wedge \begin{array}{c} Q_1 a Q_2 \\ {\scriptstyle\psi''}\Vert \\ (C_1 \vee Q_2') \end{array}}{\begin{array}{c} Q_1' \vee Q_2' \\ (1 \wedge C_1) \vee \ {\scriptstyle\psi'}\Vert \\ C_2 \end{array}}, \qquad \begin{array}{c} {\scriptstyle\phi_1}\Vert \\ A \vee Q_1 \end{array}, \qquad \begin{array}{c} {\scriptstyle\phi_2}\Vert \\ B \vee Q_2 \end{array}$$

such that $|\phi_1|_\vee + |\phi_2|_\vee \leq |\phi_2'|_\vee \leq |\phi'|_\vee \leq |\phi|_\vee$.
If ϕ is tame, then ψ', ϕ_1', ϕ_2' and ψ'' are tame. Therefore, ψ is tame. Furthermore, by induction hypothesis, ϕ_1 and ϕ_2 are equalities.

- $\phi =_\vee \boxed{\begin{array}{c} {\scriptstyle\phi'}\Vert \\ =_\vee \dfrac{\top \vee C}{(\top a \top) \vee C} \end{array}}$ with $A =_\vee B =_\vee \top$

We take:

$$\psi \equiv \wedge_{\downarrow} \dfrac{\boxed{=\dfrac{\bot a \bot}{\bot}} \ \vee \ \bot \wedge \boxed{\begin{array}{c} {\scriptstyle\phi'}\Vert \\ 1 \vee C \end{array}}}{\boxed{=_\vee \dfrac{\bot \wedge \top}{\bot}} \ \vee \ \bot \vee C}, \quad \phi_1 \equiv {}^{=_\vee}\dfrac{\top}{\boxed{=_\vee \dfrac{\top}{A}} \ \vee \ \bot}, \quad \phi_2 \equiv {}^{=_\vee}\dfrac{\top}{\boxed{=_\vee \dfrac{\top}{B}} \vee \bot}$$

with $|\phi_1|_\vee = |\phi_2|_\vee = 0$.
If ϕ is tame, ψ is tame. Furthermore, ϕ_1 and ϕ_2 are equalities.

- $\phi =_\vee \boxed{=_\vee \dfrac{\overset{\phi'}{\Vdash}\ \bot \vee C}{(\bot a \bot) \vee C}}$ with $A =_\vee B =_\vee \bot$

We take

$$\psi \equiv \wedge_\downarrow \frac{\boxed{= \frac{\top a \top}{\top}} \vee \bot \wedge \boxed{\overset{\phi'}{\Vdash}\ \bot \vee C}}{\boxed{=_\vee \frac{\top \wedge \bot}{\bot}} \vee \bot \vee C}, \quad \phi_1 \equiv \overset{=_\vee}{\frac{\top}{\boxed{=_\vee \frac{\bot}{A}} \vee \top}}, \quad \phi_2 \equiv \overset{=_\vee}{\frac{\top}{\boxed{=_\vee \frac{\bot}{B}} \vee \top}}$$

with $|\phi_1|_\vee = |\phi_2|_\vee = 0$.
If ϕ is tame, ψ is tame. Furthermore, ϕ_1 and ϕ_2 are equalities. □

Splitting tells us that from contexts where the main connective is $\vee$ we can follow occurrences of $\wedge$, ⧈ and of the atoms up in the proof and obtain independent subproofs. We can now apply splitting starting from the outermost occurrences of $\wedge$, ⧈ or the atoms, and apply this process recursively on every subproof to obtain a series of nested subproofs that in a way make-up the original proof. This recursive process is formalize in Aler Tubella's PhD thesis [1].

Definition 7.3.10 (Provable context)**.** We say that a context $H\{\}$ is *provable* if $H\{\top\} = \top$.

Definition 7.3.11 (Height of a context)**.** Given a context $S\{\}$ we define the *height of a context* as the number of instances of $\wedge$, ⧈ and a that appear in the scope of $\{\}$. We denote it by $|S|_\vee$.

Theorem 7.3.12 (Context Reduction)**.** *For any formula A and for any context $S\{\}$, given a proof* $\overset{\phi}{\underset{S\{A\}}{\Vdash}}\ \mathsf{SAK^K S_{sp}}$*, there exist a formula K, a provable context $H\{\}$ and derivations*

$$\overset{\delta}{\underset{A \vee K}{\Vdash}}\ \mathsf{SAK^K S_{sp}} \quad \textit{and} \quad \begin{array}{c} H\{\{\} \vee K\} \\ \chi \Vdash \\ S\{\} \end{array}$$

such that if ϕ is tame, then δ is tame. Furthermore, if $\{\}$ is not in the scope of an atom in $S\{\}$ and ϕ is tame, then χ' is tame.

Proof. We proceed by induction on the number of connectives $\alpha \neq \vee$ that $\{\}$ is in the scope of. We denote it by $|S|_\vee$.

- If $|S|_\vee = 0$, then $S\{A\} =_\vee A \vee K$ and we take $\chi =_\vee \phi$ and $H\{\} = \{\}$.
- If $S\{A\} =_\vee (S'\{A\} \wedge B) \vee C$, then we apply Theorem 7.3.9 to ϕ. Therefore, there exist derivations

$$\begin{array}{c} Q_1 \lor Q_2 \\ {\scriptstyle\psi}\Vert \\ C \end{array}, \qquad \begin{array}{c} \\ {\scriptstyle\phi_1}\Vert \\ S'\{A\} \lor Q_1 \end{array}, \qquad \begin{array}{c} \\ {\scriptstyle\phi_2}\Vert \\ B \lor Q_2 \end{array}$$

such that ϕ_1, ϕ_2 and ψ are tame if ϕ is tame. We apply induction hypothesis to ϕ_1 since $|S'|_\lor < |S|_\lor$. Then, there are derivations

$$\begin{array}{c} {\scriptstyle\delta}\Vert_{\mathsf{SAK^KS_{sp}}} \\ A \lor K \end{array} \text{ and } \begin{array}{c} H'\{\{\} \lor K\} \\ {\scriptstyle\chi'}\Vert \\ S'\{\} \lor Q_1 \end{array}$$

with H' a provable context, such that δ is tame if ϕ_1 is tame. We take $H\{\} = H'\{\} \land \top$. We have $H\{\top\} = H'\{\top\} \land \top = \top \land \top = \top$, and we can build in $\mathsf{SAK^KS_{sp}}$:

$$\chi \equiv \land_\downarrow \dfrac{\begin{array}{c} H'\{\{\} \lor K\} \\ {\scriptstyle\chi'}\Vert \\ S'\{\} \lor Q_1 \end{array} \land \begin{array}{c} {\scriptstyle\phi_2}\Vert \\ B \lor Q_2 \end{array}}{\begin{array}{c} Q_1 \lor Q_2 \\ (S'\{\} \land B) \lor \quad {\scriptstyle\psi}\Vert \\ C \end{array}}$$

since the connective $\land$ is not an atom, $\{\}$ is not in the scope of an atom in $H\{\}$ and χ is tame.

- If $S\{A\} =_\lor (B \land S'\{A\}) \lor C$, then we apply Theorem 7.3.9 to ϕ. Therefore, there exist derivations:

$$\begin{array}{c} Q_1 \lor Q_2 \\ {\scriptstyle\psi}\Vert \\ C \end{array}, \qquad \begin{array}{c} \\ {\scriptstyle\phi_1}\Vert \\ B \lor Q_1 \end{array}, \qquad \begin{array}{c} \\ {\scriptstyle\phi_2}\Vert \\ S'\{A\} \lor Q_2 \end{array}$$

such that ϕ_1, ϕ_2 and ψ are tame if ϕ is tame.

We apply induction hypothesis to ϕ_2 since $|S'|_\lor < |S|_\lor$. Then, there are derivations

$$\begin{array}{c} {\scriptstyle\delta}\Vert_{\mathsf{SAK^KS_{sp}}} \\ A \lor K \end{array} \text{ and } \begin{array}{c} H'\{\{\} \lor K\} \\ {\scriptstyle\chi'}\Vert \\ S'\{\} \lor Q_2 \end{array}$$

with H' a provable context, such that δ is tame if ϕ_2 is tame.

We take $H\{\} = H'\{\} \land \top$. We have $H\{\top\} = H'\{\top\} \land \top = \top \land \top = \top$, and we can build in $\mathsf{SAK^KS_{sp}}$:

$$\chi \equiv \wedge_{\downarrow} \dfrac{\begin{array}{c} \phi_1 \Big\| \\ B \vee Q_1 \end{array} \wedge \begin{array}{c} H'\{\{\} \vee K\} \\ \chi' \Big\| \\ S'\{\} \vee Q_2 \end{array}}{\begin{array}{c} Q_1 \vee Q_2 \\ (B \wedge S'\{\}) \vee \quad \psi \Big\| \\ C \end{array}}$$

since the connective $\wedge$ is not an atom, $\{\}$ is not in the scope of an atom in $H\{\}$ and χ is tame.

- If $S\{A\} =_{\vee} (S'\{A\} \boxtimes B) \vee C$, we apply Theorem 7.3.9 to ϕ. Then, there exist Q_1 and Q_2 such that

$$\begin{array}{c} Q_1 \boxtimes Q_2 \\ \psi \Big\| \\ C \end{array}, \qquad \begin{array}{c} \phi_1 \Big\| \\ S'\{A\} \vee Q_1 \end{array}, \qquad \begin{array}{c} \phi_2 \Big\| \\ B \vee Q_2 \end{array}$$

such that ϕ_1, ϕ_2 and ψ are tame if ϕ is tame. We apply induction hypothesis to ϕ_1 since $|S'|_{\vee} < |S|_{\vee}$. Then, there are derivations

$$\begin{array}{c} \delta \Big\|_{\mathsf{SAK^K S_{sp}}} \\ A \vee K \end{array} \text{ and } \begin{array}{c} H'\{\{\} \vee K\} \\ \chi' \Big\| \\ S'\{\} \vee Q_1 \end{array}$$

with H' a provable context and K a formula. We take $H\{\} = H'\{\} \boxtimes \top$. We have $H\{\top\} = H'\{\top\} \boxtimes \top = \top \boxtimes \top = \top$, and we can build in $\mathsf{SAK^K S_{sp}}$:

$$\chi \equiv \mathsf{sak}_{\downarrow} \dfrac{\begin{array}{c} H'\{\{\} \vee K\} \\ \chi' \Big\| \\ S'\{\} \vee Q_1 \end{array} \boxtimes \begin{array}{c} \phi_2 \Big\| \\ B \vee Q_2 \end{array}}{\begin{array}{c} Q_1 \boxtimes Q_2 \\ (S'\{\} \boxtimes B) \vee \quad \psi \Big\| \\ C \end{array}}$$

- If $S\{A\} =_{\vee} (S'\{A\} a B) \vee C$, we apply Theorem 7.3.9 to ϕ. Then, there exist Q_1 and Q_2 such that

$$\begin{array}{c} Q_1 \; a \; Q_2 \\ \psi \Big\| \\ C \end{array}, \qquad \begin{array}{c} \phi_1 \Big\| \\ S'\{A\} \vee Q_1 \end{array}, \qquad \begin{array}{c} \phi_2 \Big\| \\ B \vee Q_2 \end{array}$$

We apply induction hypothesis to ϕ_1 since $|S'|_{\vee} < |S|_{\vee}$. Then, there are derivations

$$\begin{array}{c} \delta \Vert \mathsf{SAK^K S_{sp}} \\ A \vee K \end{array} \text{ and } \begin{array}{c} H'\{\{\} \vee K\} \\ \chi' \Vert \\ S'\{\} \vee Q_1 \end{array}$$

with H' a provable context and K a formula. We take $H\{\} = H'\{\}\ a\ \top$. We have$H\{\top\} = H'\{\top\}\ a\ \top = \top\ a\ \top = \top$. Then we can get:

$$\chi \equiv \mathsf{sai}_{\downarrow} \dfrac{\begin{array}{c} H'\{\{\} \vee K\} \\ \chi' \Vert \\ S'\{\} \vee Q_1 \end{array} \quad a \quad \begin{array}{c} \phi_2 \Vert \\ B \vee Q_2 \end{array}}{\begin{array}{c} Q_1\ a\ Q_2 \\ (S'\{\}\ a\ B) \vee \quad \psi \Vert \\ C \end{array}}$$

□

As a corollary of splitting and context reduction we can show the admissibility of a class of up-rules [1]. The main idea is that through splitting we can separate a proof into subproofs that are independently provable. We can then combine these subproofs differently to obtain a new proof with the same conclusion.

Corollary 7.3.13 (Admissibility of cuts)**.** *For any formula A, B, C and D, any context S, a rule* r *of* $\mathsf{SAK^K S}$ *and any proof with $\alpha \neq \vee$,*

$$\phi \equiv \begin{array}{c} \phi' \Vert \mathsf{SAK^K S_{sp}} \\ S\left\{ \mathsf{r}\uparrow \dfrac{(A \alpha B) \wedge (C \alpha D)}{(A \wedge C) \alpha (B \wedge D)} \right\} \end{array},$$

there is a proof

$$\begin{array}{c} \pi \Vert \mathsf{SAK^K S_{sp}} \\ S\{(A \wedge C) \alpha (B \wedge D)\} \end{array}$$

Furthermore, if ϕ is tame, then π is tame.

Proof. Let us take the case with $\alpha = \boxtimes$. For any formula A, B, C, D, any context S and any proof:

$$\phi \equiv \begin{array}{c} \Vert \mathsf{SAK^K S_{sp}} \\ S\left\{ \mathsf{sai}_{\uparrow} \dfrac{(A \boxtimes B) \wedge (C \boxtimes D)}{(A \wedge C) \boxtimes (B \wedge D)} \right\} \end{array},$$

We apply Theorem 7.3.12 to ϕ, then there are derivations

$$\begin{array}{c} \delta \Vert \mathsf{SAK^K S_{sp}} \\ ((A \boxtimes B) \wedge (C \boxtimes D)) \vee K \end{array} \text{ and } \begin{array}{c} H\{\{\} \vee K\} \\ \chi \Vert \\ S\{\} \end{array}$$

Then, we apply Theorem 7.3.9 to δ and we get:

$$\begin{array}{c} Q_1 \vee Q_2 \\ {\scriptstyle\psi}\Vdash \\ K \end{array}, \qquad \begin{array}{c} {\scriptstyle\phi_1}\Vdash \\ (A \boxtimes B) \vee Q_1 \end{array}, \qquad \begin{array}{c} {\scriptstyle\phi_2}\Vdash \\ (C \boxtimes D) \vee Q_2 \end{array}$$

Again we apply Theorem 7.3.9 to ϕ_1 and ϕ_2. Then, we have:

$$\begin{array}{c} Q_1' \boxtimes Q_2' \\ {\scriptstyle\psi_1}\Vdash \\ Q_1 \end{array}, \qquad \begin{array}{c} {\scriptstyle\phi_1'}\Vdash \\ A \vee Q_1' \end{array}, \qquad \begin{array}{c} {\scriptstyle\phi_2'}\Vdash \\ B \vee Q_2' \end{array}$$

$$\begin{array}{c} Q_1'' \boxtimes Q_2'' \\ {\scriptstyle\psi_2}\Vdash \\ Q_2 \end{array}, \qquad \begin{array}{c} {\scriptstyle\phi_1''}\Vdash \\ C \vee Q_1'' \end{array}, \qquad \begin{array}{c} {\scriptstyle\phi_2''}\Vdash \\ D \vee Q_2'' \end{array}$$

We can then build the following proof in $\mathsf{SAK}^{\mathsf{K}}\mathsf{S}_{\mathsf{sp}}$:

$$\pi \equiv H\left\{\begin{array}{c} \mathsf{sak}_{\downarrow}\dfrac{\wedge_{\downarrow}\dfrac{\boxed{\begin{array}{c}{\scriptstyle\phi_1'}\Vdash\\ A\vee Q_1'\end{array}}\wedge\boxed{\begin{array}{c}{\scriptstyle\phi_1''}\Vdash\\ C\vee Q_1''\end{array}}}{(A\wedge C)\vee(Q_1'\vee Q_1'')}\boxtimes\wedge_{\downarrow}\dfrac{\boxed{\begin{array}{c}{\scriptstyle\phi_2'}\Vdash\\ B\vee Q_2'\end{array}}\wedge\boxed{\begin{array}{c}{\scriptstyle\phi_2''}\Vdash\\ D\vee Q_2''\end{array}}}{(B\wedge D)\vee(Q_2'\vee Q_2'')}}{(A\wedge C)\boxtimes(B\wedge D)\vee\mathsf{sak}_{\downarrow}\dfrac{(Q_1'\vee Q_1'')\boxtimes(Q_2'\vee Q_2'')}{\boxed{\begin{array}{c}\boxed{\begin{array}{c}Q_1'\boxtimes Q_2'\\ {\scriptstyle\psi_1}\Vdash\\ Q_1\end{array}}\vee\boxed{\begin{array}{c}Q_1''\boxtimes Q_2''\\ {\scriptstyle\psi_2}\Vdash\\ Q_2\end{array}}\\ {\scriptstyle\psi}\Vdash\\ K\end{array}}}}\\ {\scriptstyle\chi}\Vdash\\ S\{(A\wedge C)\boxtimes(B\wedge D)\}\end{array}\right\}$$

If ϕ is tame, then $\{\}$ is not in the scope of an atom in $S\{\}$ and $\phi_1', \phi_2', \phi_1'', \phi_2'', \psi_1, \psi_2$ and χ are tame. Therefore, π is tame.

The proof for $\alpha = a$ is similar. □

Remark 7.3.14. The rule $\vee_{\uparrow}\dfrac{(A \vee B) \wedge (C \wedge D)}{(A \wedge C) \vee (B \wedge D)}$ is admissible for systems with the rule $\wedge_{\downarrow}$ where $\wedge$ is associative. We obtain this result as follows:

$$\begin{array}{c} =\dfrac{((A \vee B) \wedge (C \wedge D)}{((A \vee B) \wedge C) \wedge D} \\ \wedge_{\downarrow}\dfrac{((A \vee B) \wedge (C \vee \bot)) \wedge D}{((A \wedge C) \vee (B \vee \bot)) \wedge D} \\ =\dfrac{((A \wedge C) \vee B) \wedge (\bot \vee D)}{(A \wedge C) \vee \bot \vee (B \wedge D)} \end{array}$$

Chapter 8

Conclusion

In this thesis, we have made a rigorous study of modal logics from the perspective of proof theory and we have presented various contributions to this field. This chapter starts with Section 8.1 where we provide a comprehensive overview of each chapter within this thesis, highlighting the principal contributions we have advanced throughout our research. Then, in Section 8.2 we outline potential next steps and areas for further study, extending the scope of our research beyond the contents of this thesis.

8.1 A tour of our thesis

We have started this thesis introducing the fundamental concepts for the develop of our contribution. In Chapter 2 and 3 we have presented the syntax and semantics of classical and intuitionistic modal logics, including Hilbert and Gentzen's axiomatizations. In both settings we have focused our attention in labelled deduction, more precisely, to labelled sequent systems due to our research interests in studying the intuitionistic approach using this formalism. In this introductory chapters, we have also presented Negri's [49] and Simpson's [59] labelled proof systems for classical and intuitionistic modal logics, respectively.

From Chapter 4, we have presented our contributions for intuitionistic modal logics. The aforementioned systems were our first understanding for the desing and development of our first contribution: our *fully labelled sequent system for intuitionistic modal logics*. This joint work with Marin and Straßburger [43] introduces a labelled sequent system to capture intuitionistic modal logics. This labelled system has not only one (as the one presented by Simpson), but two relation symbols appearing in sequents: one for the accessibility relation R associated with the Kripke semantics for normal modal logics and one for the preorder relation $\leq$ associated with the Kripke semantics for intuitionistic logic. This puts our system in close correspondence with the standard birelational Kripke semantics for intuitionistic modal logics. We have shown that this system is sound and complete, and we have presented a proof for its cut-elimination. This system was first presented for the intuitionistic modal logic IK but then, as a consequence of having the two relations explicit in the system, it can encompass a wider range of intuitionistic modal logics: in particular, we have shown extensions of the fully labelled sequent system with one-sided intuitionistic Scott-Lemmon axioms and with path Scott-Lemmon axioms, which allows us to have a completely general labelled framework to capture all the intuitionistic modal logics of the $\mathsf{S5}$-cube. This contribution is presented in Chapter 4.

The fully labelled sequent system became one of the key ingredients in addressing the *decidability problem of various intuitionistic modal logics in the* S5*-cube*. In Chapter 5, we began working on the decision problems for "simpler" logics, in the sense that their decidability was already known. In particular, we have worked first with logics where transitivity is not part of the conditions restricting the class of frames we were dealing with, as it is the case of the intuitionistic modal logics IK, IT, IKB, ITB and IK5. We have provided for them a constructive decision procedure, that, given a formula, produces either a proof showing that the formula is valid, i.e. it is a theorem of the logic, or a finite countermodel falsifying the formula, and therefore, showing that the formula is not provable in the logic. This procedure is based on our fully labelled sequent system. This result also served as a strong introduction to later delve our work into logics whose decidability were unknown until this year, as it is the case of the intuitionistic logic IS4 [21]. In particular, all the notions presented in Chapter 5 were developed with the aim of being applicable to all the intuitionistic modal logics of the S5-cube. Moreover, the search algorithm was developed with the flexibility to be customized for each specific intuitionistic modal logic we wish to employ, incorporating the necessary constraints to demonstrate its termination in each case. We have expanded this result to investigate the decidability of the logic IS4, presenting, as a result, another significant contribution of this thesis in Chapter 6.

The decision problem of the intuitionistic modal logic IS4 has been an open problem for almost thirty years since it had been posed in Simpson's PhD thesis in 1994. With Girlando, Kuznets, Marin and Straßburger [21], we have proved that the *intuitionistic logic* IS4 *is decidable*. We were able to obtained this result thanks to two key ingredients. The first one was the use of the fully labelled sequent system (as we have done for the other mentioned logics in Chapter 5), since this system inherits the advantages of labelled systems for both intuitionistic propositional logic and classical modal logics: in particular, all inference rules are invertible (i.e. we never delete information bottom-up in proof search) and there is a direct correspondence between sequents and models which lets us build a countermodel by interpreting (and extending) the labelled sequent at which proof search terminates. And second, by introducing a mechanism which gives a real proof when the search algorithm terminates in axiomatic sequents. This had to be introduced because we have solved the decision problem for IS4 by identifying (substituting) labels not only after the completion but also during proof search, and a priori this could be unsound. This means that, when terminating a branch on a non-axiomatic sequent, it is still possible to extract a countermodel from it. However, when reaching only axiomatic leaves, it remains to be shown that a sound proof can be obtained from the proof attempt (potentially containing identification of labels). For that, we have introduced the notion of *unfolding* the proof attempt into a real proof. In other words, we have shown that we can preserve soundness if we organize the proof search in a certain systematic way.

All these contributions were in the intuitionistic setting of modal logics. In Chapter 7, we have presented our work in a fragment of the classical approach. This chapter is a work in progress in the formalism of deep inference using the methodology of *subatomic proof theory*. This method treats the atoms as binary connectives, leading to a uniform shape of all inference rules. This enormously reduces the number of cases in the case analysis for cut-elimination. This arises to the question: *how this methodology can treat the modalities since they are unary connectives?*. We started exploring this new formalism and we have obtained a subatomic proof system for the linear fragment of classical modal logic. We

have proved its cut-elimination via splitting, and, as a result of this proof, we proved the admissibility of the up-rules which are the cuts. There are numerous areas for ongoing work within this system and in this field of subatomic proof theory. We are going to mention some of them in the next section.

8.2 Exploring next steps

The work we have done during this thesis leaves open several research directions that we are going to address briefly in this section.

Decision problem for IK4: as a consequence of our investigation into how the axiom of transitivity impacts the design and development of a search algorithm for establishing the decidability of a logic, we have shifted our focus towards exploring the decision problem of another intuitionistic modal logic within the S5-cube. This logic is the intuitionistic K4, known as IK4, which is analogous to IS4 but without the axiom of reflexivity. The decidability of IK4 remains as an open problem in the field (also since it had been presented by Simpson [59]).

We conjecture that the same method we applied to IS4 can also be adapted to IK4, since both require a certain treatment of the transitivity axiom as part of the logic. While we consider that the overall argument that we have implemented for IS4 will remain consistent for IK4, we anticipate encountering subtle differences in some definitions and proof arguments due to the absence of reflexivity. Consequently, a comprehensive examination of the logic IK4 has become an integral aspect of our ongoing research interests.

Implementation of the search algorithm: we are also interested in the implementation of the search algorithm presented in Chapters 5 and 6 in order to study its efficiency and to be able to produce more interesting examples. Our interest in this line of research began some time ago. With Girlando [22], we developed a prototype Prolog theorem prover implementing the fully labelled sequent calculus for IK. This prototype was called MOILab, for *MOdal and Intuitionistic Labelled sequents*[1] which was built upon MOIN [23], a theorem prover implementing nested sequent calculi (both single-conclusion and multi-conclusion) for all the logics in the modal intuitionistic cube. MOILab consists of a set of Prolog clauses, each implementing a rule of the labelled sequent calculus. The clauses are recursively applied to a given formula, constructing a proof-search tree. MOILab yields a derivation in case of proof search success, and a countermodel in case of proof search failure. The countermodel is a birelational model, and it is extracted from the upper sequent occurring a failed branch. MOILab is a prototype, meaning that, as to now, proof search does not terminate on all IK formulas. This prototype started before coming up with all the notions and details of Chapters 5 and 6, and termination was an issue back then. Now that termination is settled, we would like to continue exploring this search algorithm for the mentioned logics in this thesis but also for all the intuitionistic modal logics of the S5-cube. We would also like to investigate **the complexity of provability in the logic IS4**.

Deep inference and subatomic proof theory for modal logics: building upon our studies in deep inference formalism and subatomic logic methodology, there are several intriguing research directions we would like to delve into. On one hand, just as we have examined other proof systems, we aspire to develop a deep inference system for intuition-

[1] MOILab is available here: `http://mariannagirlando.com/MOILab.html`

istic modal logic. On the other hand, we want to continue our studies in the subatomic methodology with the goal of achieving an internal cut-elimination proof for modal logic. Cut-elimination for deep inference systems is more involved than for traditional sequent style systems. In particular, for modal logics, no cut-elimination proof that is internal to deep inference has been given so far. We aim to expand our subatomic proof system, originally devised for the linear fragment of classical modal logic, to encompass classical modal logic in its entirety and obtain a cut-elimination result. A proof of cut-elimination via splitting usually consists of two parts. Only the second one is the actual splitting and needs a "linear" system, i.e., one without weakening and contraction. To remove weakening and contraction, the first part of the cut-elimination performs a decomposition [26] or cycle elimination [2]. We need to work on these results to achieve the subatomic proof system for classical modal logic that we aim for.

Bi-Nested sequent for modal logic: after introducing a labelled sequent system in Chapter 4 to encompass intuitionistic modal logics using both accessibility and preorder relations, we subsequently initiated an investigation into the same concept within nested sequent systems. In nested and labelled sequents, extending the sequent structure with the same *one* extra element is enough to obtain sound and complete systems. As we have seen, this no longer matches the relational semantics of these logics, which requires to combine *both* relations. More importantly, it leads to deductive systems that are not entirely satisfactory; they cannot as modularly capture axiomatic extensions (or equivalently, restricted semantical conditions) and, in particular, can only provide decision procedures for some of them [59]. In an attempt to make the fully labelled sequent system (presented in Chapter 4) amenable for proof-search and decision procedures, with Marin we have started investigated a fully nested framework [42]. Despite having already established the decidability of IS4 through the labelled approach, there is merit in continuing our exploration of the nested variant. Our aim is to develop a nested sequent system that closely aligns with birelational Kripke semantics, and potentially, to investigate additional results regarding the decision problems of intuitionistic modal logics using the nested system.

Bibliography

[1] Andrea Aler Tubella. *A study of normalisation through subatomic logic*. PhD thesis, University of Bath, 2017.

[2] Andrea Aler Tubella, Alessio Guglielmi, and Benjamin Ralph. Removing cycles from proofs. In *26th EACSL Annual Conference on Computer Science Logic (CSL 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.

[3] Arnon Avron. The method of hypersequents in the proof theory of propositional non-classical logics. In *Logic: from foundations to applications: European logic colloquium*, pages 1–32. Clarendon Press, 1996.

[4] Gavin M. Bierman and Valeria de Paiva. On an intuitionistic modal logic. *Studia Logica*, 65(3):383–416, 2000.

[5] Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*, volume 53 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, Cambridge, 2001.

[6] Patrick Blackburn and Johan van Benthem. Modal logic: A semantic perspective. In *Handbook of Modal Logic*. Elsevier North-Holland, 2006.

[7] Luitzen Egbertus Jan Brouwer. Intuitionism and formalism. In *Philosophy and Foundations of Mathematics*, pages 123–138. Elsevier, 1975.

[8] Kai Brünnler. Deep sequent systems for modal logic. *Archive for Mathematical Logic*, 48(6):551–577, 2009.

[9] Kai Brünnler and Alwen F. Tiu. A local system for classical logic. In *International Conference on Logic for Programming Artificial Intelligence and Reasoning*, pages 347–361. Springer, 2001.

[10] Kaustuv Chaudhuri, Sonia Marin, and Lutz Straßburger. Modular focused proof systems for intuitionistic modal logics. In *FSCD 2016*, volume 52, pages 16:1–16:18. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.

[11] Brian F Chellas. *Modal logic: an introduction*. Cambridge university press, 1980.

[12] Roy Dyckhoff and Sara Negri. Proof analysis in intermediate logics. *Archive for Mathematical Logic*, 51(1):71–92, 2012.

[13] W. B. Ewald. Intuitionistic tense and modal logic. 51, 1986.

[14] Ronald Fagin. Generalized first-order spectra and polynomial-time recognizable sets. *Complexity of computation*, 7:43–73, 1974.

[15] Frederic B. Fitch. Intuitionistic modal logic with quantifiers. *Portugaliae Mathematica*, 7(2):113–118, 1948.

[16] Melvin Fitting. Modal proof theory. In *Studies in Logic and Practical Reasoning*, volume 3, pages 85–138. Elsevier, 2007.

[17] Gottlob Frege. *Begriffsschrift: eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*. Halle, 1879.

[18] Dov M. Gabbay. *Labelled Deductive Systems*. Clarendon Press, 1996.

[19] Gerhard Gentzen. Untersuchungen über das logische Schließen. I. *Mathematische Zeitschrift*, 39:176–210, 1934.

[20] Gerhard Gentzen. Untersuchungen über das logische Schließen. II. *Mathematische Zeitschrift*, 39:405–431, 1935.

[21] Marianna Girlando, Roman Kuznets, Sonia Marin, Marianela Morales, and Lutz Straßburger. Intuitionistic s4 is decidable. In *2023 Symposium on Logic in Computer Science (LICS2023)*, 2023.

[22] Marianna Girlando and Marianela Morales. Moilab: towards a labelled theorem prover for intuitionistic modal logics. *Women in Logic*, 2020.

[23] Marianna Girlando and Lutz Straßburger. Moin: A nested sequent theorem prover for intuitionistic modal logics. In *International Joint Conference on Automated Reasoning*, pages 398–407. Springer, 2020.

[24] Rajeev Goré, Linda Postniece, and Alwen F Tiu. On the correspondence between display postulates and deep inference in nested sequent calculi for tense logics. *Logical Methods in Computer Science*, 7, 2011.

[25] Alessio Guglielmi and Lutz Straßburger. Non-commutativity and MELL in the calculus of structures. In *International Workshop on Computer Science Logic*, pages 54–68. Springer, 2001.

[26] Alessio Guglielmi and Lutz Straßburger. A system of interaction and structure IV: The exponentials. *ACM Transactions on Computational Logic (TOCL)*, 2007.

[27] Joseph Halpern, Robert Harper, Neil Immerman, Phokion Kolaitis, Moshe Vardi, and Victor Vianu. On the unusual effectiveness of logic in computer science. *Bulletin of Symbolic Logic*, 7(2):213–236, 2001.

[28] Robert Hein and Charles Stewart. Purity through unravelling. *Structures and Deduction*, pages 126–143, 2005.

[29] David Hilbert. The foundations of mathematics. In Jean van Heijenoort, editor, *From Frege to Gödel: A Source Book in Mathematical Logic, 1879–1931*, pages 464–479. Cambridge MA: Harvard University Press, 1967.

[30] Ryo Kashima. Cut-free sequent calculi for some tense logics. *Studia Logica*, 53(1):119–136, 1994.

[31] Stuart A Kauffman. The sciences of complexity and "origins of order". In *PSA: proceedings of the biennial meeting of the philosophy of science association*, volume 1990, pages 299–322. Cambridge University Press, 1990.

[32] Oiva Ketonen. Untersuchungen zum Prädikatenkalkül. *Journal of Symbolic Logic*, 10(4), 1945.

[33] Stephen Cole Kleene. Introduction to metamathematics. *Journal of Symbolic Logic*, 19(3), 1952.

[34] Saul A. Kripke. A completeness theorem in modal logic. *The Journal of Symbolic Logic*, 24(1):1–14, 1959.

[35] Saul A. Kripke. Semantical analysis of modal logic I: Normal modal propositional calculi. *Mathematical Logic Quarterly*, 9(5-6):67–96, 1963.

[36] Richard E Ladner. The computational complexity of provability in systems of modal propositional logic. *SIAM journal on computing*, 6(3):467–480, 1977.

[37] François Lamarche. On the algebra of structural contexts. *Mathematical Structures in Computer Science*, 2001.

[38] Björn Lellmann. Linear nested sequents, 2-sequents and hypersequents. In *TABLEAUX: Automated Reasoning with Analytic Tableaux and Related Methods*, pages 135–150, 2015.

[39] Edward John Lemmon and Dana S. Scott. *An Introduction to Modal Logic*. Blackwell, Oxford, 1977.

[40] Paolo Maffezioli, Alberto Naibo, and Sara Negri. The Church–Fitch knowability paradox in the light of structural proof theory. *Synthese*, 190(14):2677–2716, 2013.

[41] Sonia Marin. *Modal proof theory through a focused telescope*. PhD thesis, Université Paris-Saclay, 2018.

[42] Sonia Marin and Marianela Morales. Fully structured proof theory for intuitionistic modal logics. In *AiML 2020-Advances in Modal Logic*, 2020.

[43] Sonia Marin, Marianela Morales, and Lutz Straßburger. A fully labelled proof system for intuitionistic modal logics. *Journal of Logic and Computation*, 31(3):998–1022, 2021.

[44] Sonia Marin and Lutz Straßburger. Label-free modular systems for classical and intuitionistic modal logics. In Rajeev Goré and Agi Kurucz, editors, *Advances in Modal Logic 10, Groningen*, pages 387–406, 2014.

[45] Sonia Marin and Lutz Straßburger. Proof theory for indexed nested sequents. In Renate A. Schmidt and Cláudia Nalon, editors, *Automated Reasoning with Analytic Tableaux and Related Methods: 26th International Conference, TABLEAUX 2017, Brasília, Brazil, September 25–28, 2017, Proceedings*, pages 81–97. Springer International Publishing, 2017.

[46] Andrea Masini. 2-sequent calculus: A proof theory of modalities. *Annals of Pure and Applied Logic*, 58:229–246, 1992.

[47] Grigorii Efroimovich Mints. Some calculi of modal logic. *Trudy Matematicheskogo Instituta imeni VA Steklova*, 98:88–111, 1968.

[48] Sara Negri. Contraction-free sequent calculi for geometric theories with an application to barr's theorem. *Archive for Mathematical Logic*, 42(4):389–401, 2003.

[49] Sara Negri. Proof analysis in modal logic. *Journal of Philosophical Logic*, 34(5):507–544, 2005.

[50] Sara Negri. Proof theory for modal logic. *Philosophy Compass*, 6(8):523–538, 2011.

[51] Katudi Ono. Logische untersuchungen Über die grundlagen der mathematik. *Journal of Symbolic Logic*, 4(2):89–90, 1938.

[52] Gordon D. Plotkin and Colin P. Stirling. A framework for intuitionistic modal logic. In J. Y. Halpern, editor, *1st Conference on Theoretical Aspects of Reasoning About Knowledge*. Morgan Kaufmann, 1986.

[53] Francesca Poggiolesi. The method of tree-hypersequents for modal propositional logic. In *Towards Mathematical Philosophy*, volume 28 of *Trends in Logic*, pages 31–51. Springer, 2009.

[54] John Reynolds. Three approaches to type structure. In *Mathematical Foundations of Software Development: Proceedings of the International Joint Conference on Theory and Practice of Software Development (TAPSOFT) Berlin, March 25–29, 1985 Volume 1: Colloquium on Trees in Algebra and Programming (CAAP'85) 10*, pages 97–138. Springer, 1985.

[55] John A. Robinson. Computational logic: The unification computation. *Machine intelligence*, 6:63–72, 1971.

[56] Masahiko Sato. A study of Kripke-type models for some modal logics by Gentzen's sequential method. *Publications of the RIMS*, 13(2):381–468, 1977.

[57] Dana Scott. A decision method for validity of sentences in two variables. *Journal of Symbolic Logic*, 27(377):74, 1962.

[58] Gisèle Fischer Servi. Axiomatizations for some intuitionistic modal logics. *Rendiconti del Seminario Matematico dell' Università Politecnica di Torino*, 42(3):179–194, 1984.

[59] Alex Simpson. *The Proof Theory and Semantics of Intuitionistic Modal Logic*. PhD thesis, University of Edinburgh, 1994.

[60] Lutz Straßburger. Cut elimination in nested sequents for intuitionistic modal logics. In Frank Pfenning, editor, *FoSSaCS'13*, volume 7794 of *LNCS*, pages 209–224. Springer, 2013.

[61] Lutz Straßburger and Roman Kuznets. Maehara-style modal nested calculi. *Archive for Mathematical Logic*, 58(3-4):359–385, 2019.

[62] Alwen Tiu, Linda Postniece, and Rajeev Gore. On the correspondence between display postulates and deep inference in nested sequent calculi for tense logics. *Logical Methods in Computer Science*, 7, 2011.

[63] Luca Viganò. Labelled non-classical logics. *Kluwer Academic Publisher*, 2000.

[64] Georg von Wright. *An Essay in Modal Logic*. Amsterdam, North-Holland Pub. Co., 1951.

[65] Heinrich Wansing. *Sequent Systems for Modal Logics*, volume 8 of *In Dov M. Gabbay and Franz Guenthner, Handbook of Philosophical Logic*. Springer, 2002.

[66] Duminda Wijesekera. Constructive modal logics I. *Annals of Pure and Applied Logic*, 50(3):271–301, 1990.

ECOLE DOCTORALE

Titre : Systèmes de preuve inhabituels pour la logique modale avec applications aux problèmes de décision

Mots clés : Logiques modales ; Théorie de la preuve ; Décidabilité

Résumé : Cette thèse explore rigoureusement les logiques modales, les analysant du point de vue de la théorie de la preuve. Dans le contexte intuitionniste des logiques modales, nous présentons un *système de séquents entièrement étiqueté*, dans lequel non pas un, mais deux symboles de relation apparaissent dans les séquents : l'un pour la relation d'accessibilité associée à la sémantique de Kripke des logiques modales normales, et l'autre pour la relation de préordre associée à la sémantique de Kripke de la logique intuitionniste. Nous étendons ce système pour englober les quinze logiques modales intuitionnistes distinctes du cube S5. Les avantages de ce système, tels que toutes ses règles d'inférence sont inversibles et qu'il existe une correspondance directe entre les séquents et les modèles, nous permettent d'explorer les problèmes de décision pour différentes variantes des logiques modales intuitionnistes, telles que IK, IT, IB, ITB, IK5. En particulier, nous effectuons une recherche de preuves dans notre système de séquents entièrement étiqueté et nous proposons une procédure de décision pour prouver la décidabilité des logiques mentionnées. Nous étendons ensuite cet algorithme de recherche pour résoudre un problème ouvert depuis près de trente ans depuis qu'il a été posé dans la thèse de doctorat de Simpson en 1994 : nous démontrons la *décidabilité de la logique modale intuitionniste S4*. Notre algorithme de recherche produit soit une preuve, soit un contre-modèle fini, établissant ainsi la propriété de modèle fini pour l'intuitionniste S4, qui était un autre problème ouvert de longue date dans ce domaine. Dans le cadre classique des logiques modales, nous présentons un travail en cours dans le formalisme de l'inférence profonde en utilisant la méthodologie de la théorie de la preuve subatomique. Nous présentons un système de preuve subatomique pour le fragment linéaire des logiques modales classiques, montrant son *cut-elimination via splitting*. Suite à cette preuve, nous démontrons l'admissibilité des règles ascendantes, qui sont les *cuts*.

Title : Unusual proof systems for modal logics with applications to decision problems

Keywords : Modal Logics ; Proof Theory ; Decidability

Abstract : This thesis rigorously explores modal logics, analyzing them from a proof theory perspective. In the intuitionistic setting of modal logics, we present a *fully labelled sequent system* such that there is not only one, but two relation symbols appearing in sequents : one for the accessibility relation associated with the Kripke semantics for normal modal logics and one for the preorder relation associated with the Kripke semantics for intuitionistic logic. We extend this system to capture the fifteen distinct intuitionistic modal logics of the S5-cube. The advantages of this system, such as all its inference rules are invertible and that there is a direct correspondence between sequents and models, allow us to explore decision problems for different variants of intuitionistic modal logics, such as IK, IT, IB, ITB, IK5. In particular, we perform proof search in our fully labelled sequent system and we give a decision procedure to prove decidability of the mentioned logics. We then extend this search algorithm to solve a problem that has been open for almost thirty years since it had been posed in Simpson's PhD thesis in 1994 : we demonstrate *decidability of the intuitionistic modal logic S4*. Our search algorithm outputs either a proof or a finite countermodel, thus, additionally establishing the finite model property for intuitionistic S4, which has been another long-standing open problem in the area. In the classical setting of modal logics, we present a work in progress in the formalism of deep inference using the methodology of *subatomic proof theory*. We present a subatomic proof system for the linear fragment of classical modal logics, showing its cut-elimination via splitting. As a result of this proof, we show the admissibility of the up-rules which are the cuts.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France