Proof theory for fragments of the modal mu-calculus

MSc Thesis *(Afstudeerscriptie)*

written by

Lukas Zenger
(born 31.12.1994 in Belp BE, Switzerland)

under the supervision of **Dr. Bahareh Afshari**, and submitted to the Examinations Board in partial fulfillment of the requirements for the degree of

MSc in Logic

at the *Universiteit van Amsterdam.*

Date of the public defense:	**Members of the Thesis Committee:**
18.02.2021	Prof. Dr. Yde Venema (Chair)
	Dr. Bahareh Afshari (Supervisor)
	Dr. Benno van den Berg
	Prof. Dr. Rosalie Iemhoff

INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

Abstract

In this thesis we investigate the proof theory of the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$ of the modal mu-calculus. This fragment consists of formulas which have syntactic fixed point alternation depth of at most one. $\Sigma_1^\mu \cup \Pi_1^\mu$ contains the building blocks for interesting concepts such as common knowledge. Moreover, it is computationally important in view of applications in database theory. We define a circular proof system and a circular tableaux system for $\Sigma_1^\mu \cup \Pi_1^\mu$ and prove soundness and completeness. We then use these systems to establish key properties of $\Sigma_1^\mu \cup \Pi_1^\mu$, such as the finite model property and Craig interpolation. Furthermore, we define infinitary proof systems for the whole modal mu-calculus and show that they are sound and complete. The main contribution of the thesis is an axiomatization of $\Sigma_1^\mu \cup \Pi_1^\mu$ as well as novel proofs of the finite model property and Craig interpolation.

Acknowledgements

I could not have written this thesis without all the support I received in the process. I would like to take the opportunity to express my gratitude towards everyone, who was involved. First and foremost to Bahareh, for supervising my thesis and for the effort and time you devoted to teach me. Most of what I know about proof theory and the modal mu-calculus, I have learned from you. I would also like to thank you for your advice in choosing my PhD position and the role you played in my academic growth over the past two years overall. I hope we may continue our collaboration in the future. A special thanks goes to George. I have benefited a great deal from your knowledge and experience and I appreciate everything you have done for me in the past years. Your suggestion to continue my studies at the ILLC has lead to one of the best decisions in my life. Thank you to Beatrice and Christoph for all your support, not only in the last two years, but throughout my whole life. And thank you for enabling me to chase my dreams and study in Amsterdam. Last but not least, I would like to thank the many people I met at the ILLC in the last two years. Tianwei, for all your love and for turning a difficult year into a wonderful one. It was my lucky day when I met you. Freddy, Bas and Simon, for your friendship and the incredible time we spent together. It has been one of the best in my life.

Contents

1 Introduction **1**
1.1 Contributions . . . 2
1.2 Outline of the thesis . . . 3

2 The modal mu-calculus **4**
2.1 Introduction . . . 4
2.2 Syntax . . . 5
2.3 Semantics . . . 7
2.4 The alternation depth hierarchy . . . 9

3 Tableaux, proof systems and model checking games **12**
3.1 Introduction . . . 12
3.2 Model checking games . . . 13
3.3 The tableaux system **T** . . . 17
3.4 Soundness of **T** . . . 22

4 Finite model property **26**
4.1 Introduction . . . 26
4.2 Finite model property for Σ_1^μ . . . 27
4.3 Circular tableaux for $\Sigma_1^\mu \cup \Pi_1^\mu$ . . . 30
4.4 Finite model property for $\Sigma_1^\mu \cup \Pi_1^\mu$ . . . 34

5 Infinitary proof systems for the modal mu-calculus **42**
5.1 Introduction . . . 42
5.2 The sequent calculus **DT** . . . 43
5.3 Soundness and completeness of **DT** . . . 44
5.3.1 Preliminaries . . . 44
5.3.2 Completeness of **DT** . . . 50
5.3.3 Soundness of **DT** . . . 54
5.4 The sequent calculus **DT**$'$ . . . 59
5.5 Soundness and completeness of **DT**$'$ . . . 60

5.6 The two-sided sequent calculus **2DT** . . . 63
5.7 Soundness and completeness of **2DT** . . . 66

6 Craig interpolation **76**
6.1 Introduction . . . 76
6.2 The circular sequent calculus **C2DT** . . . 77
6.2.1 Soundness and completeness of **C2DT** . . . 80
6.3 Craig interpolation for Σ_1^μ and Π_1^μ . . . 82
6.4 Craig interpolation for $\Sigma_1^\mu \cup \Pi_1^\mu$ . . . 89
6.5 Optimizing Craig interpolation . . . 91

7 Discussion and further research **98**

Chapter 1

Introduction

The logical system investigated in this thesis is the modal mu-calculus, introduced by Kozen in 1983 [12]. The modal mu-calculus is an extension of propositional modal logic with fixed point operators, namely the least fixed point operator μ and the greatest fixed point operator ν. The resulting system is not only very expressive, but enjoys many desirable logical properties and has important applications in computer science. An important concept in the theory of the modal mu-calculus is the notion of fixed point alternation, which counts the number of alternations of least and greatest fixed point operators in a formula. Fixed point alternation substantially increases the expressive power of the modal mu-calculus but also the difficulty of its mathematical theory [5]. Our interest concerns a specific fragment of the modal mu-calculus which is called the first level of the alternation depth hierarchy, denoted by $\Sigma_1^\mu \cup \Pi_1^\mu$. This fragment consists of formulas that contain syntactic fixed point alternation of at most one. The interest in this fragment is motivated by two main reasons. First, the fragment contains the building blocks of interesting concepts such as common knowledge, a concept which is extensively used in epistemic logic. Moreover, $\Sigma_1^\mu \cup \Pi_1^\mu$ can be regarded as the starting point for an investigation of the alternation free fragment of the modal mu-calculus. Second, the mathematical theory of this fragment has not yet been studied and little is known about its logical properties. We aim to contribute to the investigation of this fragment by constructing circular proof systems for $\Sigma_1^\mu \cup \Pi_1^\mu$ and use them to establish that the fragment enjoys both the finite model property and Craig interpolation. While these results have already been established for the whole modal mu-calculus (see [17] and [7]), we hope to provide much simpler proofs for the first level of the alternation depth hierarchy and thereby deepen our understanding of it. Moreover, we study infinitary Gentzen style proof systems and provide soundness and completeness results. These infinitary systems are used to obtain circular proof systems for $\Sigma_1^\mu \cup \Pi_1^\mu$ and build in that sense the basis of our investigations.

There are two standard approaches to infinitary Gentzen style proof systems for the modal mu-calculus [18], that differ in the type of rules which are used for fixed point operators. The first approach is characterized by infinite unfolding of fixed point formulas, which results in pre-proofs being finite branching trees containing infinite branches. Whether a pre-proof is a proof is then decided by checking certain conditions imposed on infinite branches. The first such system was proposed by Niwiński and Walukiewicz in 1996 [16] in form of a tableaux system. The second type was developed by Jäger, Kretz and Studer [10] in 2008 and is characterised by approximating fixed points. Instead of unfolding fixed point formulas infinitely often, one derives infinitely many approximations of the fixed point and then uses a so-called ω-rule which takes all of the infinitely many approximations as premises and infers the fixed point formula. This implies that proofs in this setting are infinite branching trees. As each approximation is itself finite, every branch of such a proof-tree is finite. For an overview of the connection between these two types of systems, we refer to [18]. The proof systems developed in this thesis are of the first type and use fixed point unfolding rules. We construct in total three different but closely related infinitary sequent calculi which are sound and complete. The starting point of the construction is thereby the infinitary tableaux system developed by Niwiński and Walukiewicz in [16], which we dualize in a first step into a Gentzen style sequent calculus. This dualized proof system builds the foundation of the other two systems.

Circular proof systems for the modal mu-calculus were introduced by Jungteerapanich [11] in 2009 and more recently by Afshari and Leigh in [1]. Circular proofs have a close connection to regular infinitary proofs. An infinitary proof in our setting is a finite branching tree that contains infinite branches. Such a tree is called regular if it is the unfolding of a finite tree. Given a finite tree that unfolds into a regular tree, this finite tree is turned into a circular proof tree by adding loops to some of its leafs (hence the name 'circular'). That is, circular proof trees are essentially finite trees that unfold into infinite regular trees over their loops. In order to ensure that a circular proof system is sound, one imposes conditions on the finite proof trees that ensure that their unravelling is indeed an infinite proof. Proving the existence of circular proofs coincides with finding appropriate finite structures in infinite proof trees, which can be unfolded into regular trees. In the presence of arbitrary fixed point alternation, such a task is tricky, as the fixed point alternation makes it difficult to impose conditions on circular proofs that ensure that the system is sound. In the presence of syntactic fixed point alternation of at most one however, we show that finding appropriate finite structures is much easier. That is, we show how to define sound and complete circular proof systems for the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$. As derivations in a circular proof system are finite, we show how to use such systems to establish both the finite model property and Craig interpolation.

1.1 Contributions

The main contribution of the thesis is the construction of a circular tableaux system and a circular proof system for the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$. With these systems we provide - as far as we know - the first axiomatization of this fragment. Moreover, the systems are essential for establishing the finite model property and Craig interpolation. While these two properties are already known for the whole modal mu-calculus, we provide novel proofs for $\Sigma_1^\mu \cup \Pi_1^\mu$ which are much simpler than the proofs for the whole calculus. In doing so we hope to provide new insights into the fragment. Apart from circular systems, our second contribution is the discussion of infinitary Gentzen style sequent calculi for the whole modal mu-calculus. The infinitary systems presented are not essentially new (indeed other authors have used similar systems, see for example [18]), but we do provide novel soundness and completeness proofs by using the connection between the infinitary proof systems and the infinitary tableaux system from [16]. These proofs also provide new insights into the connection between tableaux and proof systems.

1.2 Outline of the thesis

The next two chapters lay the foundations for the rest of the thesis. They present standard results and definitions of the modal mu-calculus. The remaining parts from chapter 4 on consist of the research contributions of this thesis.

- ▷ Chapter 2 consists of a brief introduction to the modal mu-calculus. We introduce its syntax and semantics and define the alternation depth hierarchy.
- ▷ Chaper 3 introduces the tableaux system **T** developed by Niwiński and Walukiewicz in [16]. Moreover, model checking games are introduced and the soundness proof of **T** is discussed.
- ▷ Chapter 4 introduces the circular tableaux system **CT**, establishes its soundness and completeness with respect to $\Sigma_1^\mu \cup \Pi_1^\mu$ and derives as a corollary the finite model property for $\Sigma_1^\mu \cup \Pi_1^\mu$.
- ▷ Chapter 5 defines and discusses the three infinitary sequent calculi $\mathbf{DT}, \mathbf{DT}'$ and **2DT**. The chapter also consists of soundness and completeness proofs for all three systems.
- ▷ Chapter 6 introduces the circular sequent calculus **C2DT**, establishes its soundness and completeness with respect to $\Sigma_1^\mu \cup \Pi_1^\mu$ and then establishes the Craig interpolation property for $\Sigma_1^\mu \cup \Pi_1^\mu$. The last part of this chapter is devoted to discuss the optimization of the constructed interpolant.
- ▷ Chapter 7 consists of a short discussion of the established results and poses several remaining open questions which might be tackled in further research.

Chapter 2

The modal mu-calculus

2.1 Introduction

The propositional modal mu-calculus was introduced by Kozen in 1983 [12]. It is an extension of propositional modal logic with a least and a greatest fixed point operator. This creates a logical system that far exceeds the expressive power of modal logic. The modal operator $\Box$ used in modal logic provides quantification over neighbours of the current state. The formula $\Box P$ expresses that the condition P holds in every state which is reachable from the current state over a single transition step. Adding fixed point operators introduces concepts such as path-quantification. For instance, one can express the following statement:

"The condition P holds in every state reachable over an arbitrary number of transition steps."

Here, the quantification is no longer local but ranges over every path through the transition system starting in the current state. While path quantification is a much stronger form of quantification than what is provided in modal logic, it is only a weak concept compared to what is expressible in the modal mu-calculus. Apart from its expressive power, the modal mu-calculus enjoys many desirable logical properties such as decidability [5], a property which is lost in other expressive systems such as first-order logic. For a discussion why modal logics in general and the modal mu-calculus in particular are robustly decidable, we refer to [9]. Another interesting result states that the modal mu-calculus is the bisimulation invariant fragment of second-order logic, similar to modal logic, which is the bisimulation invariant fragment of first-order logic [5]. It is hence a system of considerable mathematical interest. The main application of the mu-calculus is in computer science. In the past decades, fixed point logics in general have gained a lot of attention in computer science, as they are used to specify properties of programs in the field of software verification [5]. Famous fixed point logics include Propositional Dynamic Logic (PDL), Linear Time Logic (LTL) and Computational Tree Logic (CTL), all of which are fragments of the modal mu-calculus. Indeed, many fixed point logics turn out to be included in the modal mu-calculus [5], which makes it an interesting system to study as a meta-theory. The least and greatest fixed point operators of the modal mu-calculus are not only responsible for the expressive power of the system, but also substantially increase the difficulty of its theory. Moreover, due to these operators, formulas of the modal mu-calculus are hard to grasp. In contrast to handier fixed point logics such as LTL, one requires experience and good intuition to understand what property a formula expresses. It is therefore important to obtain a good understanding of the modal mu-calculus before delving into the theory presented later on. This chapter contributes to that aim by introducing the modal mu-calculus formally. First and foremost, the syntax of the modal mu-calculus is defined in section 2.2. The subsequent section 2.3 introduces the semantics of the modal mu-calculus in terms of transition systems. The last section 2.4 of this chapter is devoted to introduce and discuss the alternation depth hierarchy and to define the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$.

For a more detailed introduction to the modal mu-calculus, we refer to the excellent overview by Bradfield and Stirling in [5] and the detailed introduction by Demri, Goranko and Lange in [8]. The presentation of this chapter closely follows the lecture notes of the course *Logic, Games and Automata* [2] taught by Afshari at the University of Amsterdam in the spring semester 2020.

2.2 Syntax

Defining the syntax of a logic starts by providing the language which is used. Throughout the thesis, we denote the language of the modal mu-calculus by L_μ.

Definition 2.2.1. The *language* L_μ of the modal mu-calculus consists of the following primitive symbols:

- ▷ A countable set of *atomic propositions* $Prop$. Atoms in $Prop$ are denoted by P and Q, possibly with sub- or superscript.
- ▷ A countable set of *variables* Var. Variables in Var are denoted by X, Y or Z, possibly with sub- or superscript.
- ▷ The logical connectives $\neg$ (*negation*), $\wedge$ (*conjunction*) and $\vee$ (*disjunction*).
- ▷ The modal operators $\Box$ (*box*) and $\Diamond$ (*diamond*).
- ▷ The fixed point operators ν (called the *greatest fixed point operator*) and μ (called the *least fixed point operator*).

When it comes to applications, it is standard to add a finite set of agent symbols A to the language of the modal mu-calculus and, instead of having a single box and diamond operator, there are modal operators for each agent $a \in A$, usually written as $[a]$ and $\langle a \rangle$. In this thesis, we deal with a single box and diamond operator, as we are interested in proof theoretic aspects of the modal mu-calculus rather than applications. The set of *literals* is defined to be $Prop \cup \{\neg P | P \in Prop\}$ and is denoted by Lit.

Definition 2.2.2. *L_μ-formulas* are defined inductively as follows:

1. If $P \in Prop$, then P and $\neg P$ are L_μ-formulas.
2. If $Z \in Var$, then Z and $\neg Z$ are L_μ-formulas.
3. If φ and ψ are L_μ-formulas, then so are $\varphi \wedge \psi$, $\varphi \vee \psi$, $\Box\varphi$ and $\Diamond\varphi$.
4. If φ is a L_μ-formula, then $\nu Z.\varphi$ and $\mu Z.\varphi$ are L_μ-formulas, provided that Z does not occur negated in φ.

Observe that negation is only applied to atoms and variables. It is more standard to present formulas of L_μ by allowing to apply negation to arbitrary formulas. Formulas as defined here are sometimes called *formulas in positive form* [2]. It is a well-known result that every formula of L_μ is equivalent to a formula in positive form, which justifies the definition presented above. If every occurrence of the variable Z in a formula φ occurs non-negated, then Z is called *positive in φ*. Given a formula of the form $\nu Z.\varphi$, we call the occurrences of Z in φ *bounded*. Occurrences of variables which are not bounded are called *free*. It follows from the definition of the semantics in the next section that if φ does not contain the variable Z, then φ is equivalent to $\sigma Z.\varphi$ for $\sigma \in \{\mu, \nu\}$. We assume from now on that whenever we deal with a formula of the form $\sigma Z.\varphi$, that Z occurs in φ. We write $\varphi(Z)$ to denote that Z occurs freely in φ. Given a formula $\sigma Z.\varphi(Z)$, the variable Z is called a *μ-variable* if $\sigma = \mu$ and it is called a *ν-variable* if $\sigma = \nu$. We stipulate that fixed point operators have higher precedence than the Boolean connectives $\wedge$ and $\vee$ which in turn have higher precedence than modal operators. That is, the formula $\nu Y.Y \wedge P$ is read as $\nu Y.(Y \wedge P)$ and the formula $\Box P \vee Q$ is read as $(\Box P) \vee Q$.

Convention 2.2.3. Given a formula $\varphi(Z)$ with Z occurring freely in φ and a formula ψ, then $\varphi(\psi)$ denotes the formula $\varphi(Z)$ where each free occurrence of Z is substituted by ψ.

In later chapters, we restrict our attention to L_μ-formulas that are closed and in guarded normal form.

Definition 2.2.4. Let φ be a L_μ-formula.

- ▷ φ is *closed* if it contains no free variables.
- ▷ φ is in *normal form* if all variables occurring in φ that are bound by different occurrences of fixed point operators are pairwise distinct.
- ▷ A variable Z is *guarded* in φ if every bound occurrence of Z in φ occurs in the scope of a modal operator. A formula φ is *guarded*, if every variable in φ is guarded.

Observe that every variable in a closed formula occurs positive.

Definition 2.2.5. Let φ be a L_μ-formula. The *set of subformulas* of φ, written $Sub(\varphi)$, is defined by induction on φ:

1. If $\varphi = P$ for $P \in Prop$, then $Sub(\varphi) := \{P\}$.
2. If $\varphi = \neg P$ for $P \in Prop$, then $Sub(\varphi) := \{\neg P\}$.
3. If $\varphi = Z$ for $Z \in Var$, then $Sub(\varphi) := \{Z\}$.
4. If $\varphi = \neg Z$ for $Z \in Var$, then $Sub(\varphi) = \{\neg Z\}$.
5. If $\varphi = \psi_1 \circ \psi_2$ for $\circ \in \{\wedge, \vee\}$, then $Sub(\varphi) := Sub(\psi_1) \cup Sub(\psi_2) \cup \{\psi_1 \circ \psi_2\}$.
6. If $\varphi = \bullet\psi$ for $\bullet \in \{\Box, \Diamond\}$, then $Sub(\varphi) := Sub(\psi) \cup \{\bullet\psi\}$.
7. If $\varphi = \sigma Z.\psi(Z)$ for $\sigma \in \{\mu, \nu\}$, then $Sub(\varphi) := Sub(\psi(Z)) \cup \{\sigma Z.\psi(Z)\}$.

If $\psi \in Sub(\varphi)$, then ψ is called a *subformula* of φ.

2.3 Semantics

Given a set S, its power set is denoted by $\mathcal{P}(S)$. Formulas of the modal mu-calculus are evaluated in transition systems.

Definition 2.3.1. A *transition system* is a triple $T = (S, \to, \rho)$ where

- ▷ S is a non-empty set; an element $s \in S$ is called a *state*
- ▷ $\to \subseteq S \times S$ is a binary *transition relation*; we write $s \to t$ for $(s,t) \in \to$
- ▷ $\rho : Prop \longrightarrow \mathcal{P}(S)$ is a function that maps atomic propositions to subsets of S

Transition systems are also known as Kripke models. Given a transition system $T = (S, \to, \rho)$, a function $V : Var \longrightarrow \mathcal{P}(S)$ that maps variables onto subsets of S is called a *valuation*. Given a transition system $T = (S, \to, \rho)$ and a valuation $V : Var \longrightarrow \mathcal{P}(S)$, we assign to each formula φ a set of states $[\![\varphi]\!]_V^T \subseteq S$, called the *truth set* of φ, with the intended meaning that φ holds at every state in $[\![\varphi]\!]_V^T$. The definition of the truth set for an atom coincides with the set of states assigned to the atom by the function ρ. Similarly, the truth set of a variable coincides with the set of states assigned to it by the valuation V. The definition of the truth sets for the Boolean connectives and the modal operators are standard. For the definition of the truth sets for fixed point formulas, suppose that some formula $\varphi(Z)$ contains a free variable Z. Then $\varphi(Z)$ induces a function $f_\varphi : \mathcal{P}(S) \longrightarrow \mathcal{P}(S)$ defined as follows:

$$f_\varphi(U) := [\![\varphi(Z)]\!]_{V[Z \mapsto U]}^T$$

where $V[Z \mapsto U]$ is defined by

$$V[Z \mapsto U](X) := \begin{cases} U \text{ if } X = Z \\ V(X) \text{ otherwise} \end{cases}$$

We call a set $U \subseteq S$ such that $f_\varphi(U) = U$ a *fixed point* of f_φ. Moreover, U is called the *greatest fixed point* of f_φ, if U is a fixed point and for every other fixed point V of f_φ it holds that $V \subseteq U$. Similarly, U is called the *least fixed point* of f_φ, if U is a fixed point and for every other fixed point V of f_φ it holds that $U \subseteq V$. If $U \subseteq f_\varphi(U)$, then U is called a *post-fixed point* of f_φ and if $f_\varphi(U) \subseteq U$, then U is called a *pre-fixed point* of $f\varphi$. The least and greatest fixed point operators are interpreted as the least and greatest fixed points of such functions. That is, the truth set assigned to the formula $\nu Z.\varphi(Z)$ is the greatest fixed point of the function f_φ. As f_φ is monotone in Z on $\mathcal{P}(S)$ due to the restriction that the variable Z only occurs non-negated, the Knaster-Tarski-Theorem [19] establishes that f_φ has a fixed point. Moreover, it is a well-known result that for such functions there exists a unique least fixed point which coincides with the intersection over all the pre-fixed points of the function and a unique greatest fixed point which coincides with the union over all its post-fixed points.

Definition 2.3.2. Let $T = (S, \rightarrow, \rho)$ be a transition system and $V : Var \longrightarrow \mathcal{P}(S)$ a valuation. The *truth set* $[\![\varphi]\!]_V^T \subseteq S$ is defined by induction on φ as follows:

$$\begin{array}{rcl}
[\![P]\!]_V^T & := & \rho(P) \\
[\![\neg P]\!]_V^T & := & S - \rho(P) \\
[\![Z]\!]_V^T & := & V(Z) \\
[\![\neg Z]\!]_V^T & := & S - V(Z) \\
[\![\varphi \wedge \psi]\!]_V^T & := & [\![\varphi]\!]_V^T \cap [\![\psi]\!]_V^T \\
[\![\varphi \vee \psi]\!]_V^T & := & [\![\varphi]\!]_V^T \cup [\![\psi]\!]_V^T \\
[\![\Box\varphi]\!]_V^T & := & \{s \in S | \text{ for all } t \in S, \text{ if } s \rightarrow t, \text{ then } t \in [\![\varphi]\!]_V^T\} \\
[\![\Diamond\varphi]\!]_V^T & := & \{s \in S | \text{ there exists } t \in S \text{ with } s \rightarrow t \text{ and } t \in [\![\varphi]\!]_V^T\} \\
[\![\nu Z.\varphi(Z)]\!]_V^T & := & \bigcup\{U \subseteq S | U \subseteq [\![\varphi(Z)]\!]_{V[Z\mapsto U]}^T\} \\
[\![\mu Z.\varphi(Z)]\!]_V^T & := & \bigcap\{U \subseteq S | [\![\varphi(Z)]\!]_{V[Z\mapsto U]}^T \subseteq U\}
\end{array}$$

If $s \in [\![\varphi]\!]_V^T$, we say that φ *holds* or equivalently is *true* at the state s of the transition system $T = (V, \rightarrow, \rho)$ under the valuation V and we call T a *model* for φ. We also write $T, V, s \models \varphi$ instead of $s \in [\![\varphi]\!]_V^T$.

Observe that the truth set of a greatest fixed point formula $\nu Z.\varphi$ is exactly the union over all post-fixed points of the function induced by φ and the truth set of a least fixed point formula is the intersection over all pre-fixed points.

Definition 2.3.3. Let φ be a formula of L_μ.

- ▷ φ is *satisfiable* if there exists a transition system $T = (S, \rightarrow, \rho)$, a valuation $V : Var \longrightarrow \mathcal{P}(S)$ and a state $s \in S$, such that $s \in [\![\varphi]\!]_V^T$.

- ▷ φ is *unsatisfiable* if it is not satisfiable.
- ▷ φ is *valid* if for every transition system $T = (S, \rightarrow, \rho)$ and every valuation V it holds that $[\![\varphi]\!]_V^T = S$.
- ▷ Two L_μ-formulas φ and ψ are called *equivalent* - written $\varphi \equiv \psi$ - if for every transition system $T = (S, \rightarrow, \rho)$ and every valuation $V : Var \longrightarrow \mathcal{P}(S)$ it holds that $[\![\varphi]\!]_V^T = [\![\psi]\!]_V^T$.

We finish this section by stating two well-known results about the modal mu-calculus which are of importance for the thesis. The first result justifies the restriction towards guarded formulas in normal form.

Proposition 2.3.4. Every L_μ-formula φ is equivalent to a guarded formula in normal form.

It is therefore safe to assume that whenever we consider an arbitrary L_μ-formula φ, that φ is guarded and in normal form. The second result is a standard equivalence which is used throughout the thesis without further mentioning.

Proposition 2.3.5. For a formula $\sigma X.\varphi(X)$ where $\sigma \in \{\mu, \nu\}$ the following holds:

$$\sigma X.\varphi(X) \equiv \varphi(\sigma X.\varphi(X))$$

The proof is based on the definition of the truth set of $\sigma X.\varphi(X)$ being a fixed point of the function induced by φ. This equivalence is of importance for the proof systems we discuss later on. Indeed it is this equivalence that motivates the rules for fixed point operators.

2.4 The alternation depth hierarchy

The expressive power of the modal mu-calculus mainly stems from fixed point alternation [5]. This alternation is defined in terms of the alternation depth hierarchy, a strictly ordered hierarchy of classes of formulas. The general idea to determine the alternation depth of a formula is to count the alternations of least and greatest fixed point operators which are in the scope of each other. However, the proper definition of fixed point alternation is a bit more involved than simply counting syntactic fixed point alternation. To see why, consider the formula *always eventually* versus the formula *infinitely often* (this example is from [5]). The formula *always eventually* is given as follows:

$$\nu Y.(\mu Z.P \vee \Diamond Z) \wedge \Diamond Y$$

The syntactic fixed point alternation depth is 2. However, computing whether this formula holds in a state of a given structure is not harder than computing two disjoint fixed points. This is because the inner fixed point is independent from the outer one and for computing the whole formula, one only has to compute the inner fixed point once. The formula *infinitely often* is given by

$$\nu Y.\mu Z.(P \vee \Diamond Z) \wedge \Diamond Y$$

Computing this formula leads to much higher complexity, as the inner fixed point now depends on the outer one.[1] Therefore, it does not suffice to simply count syntactic alternations of fixed point operators. In this section, we present the fixed point alternation depth hierarchy in the way Niwiński presented it in [15], which takes the above phenomena into account. For a more detailed introduction to the fixed point alternation hierarchy and its relevance for the modal mu-calculus, we refer to [15] and [5].

A L_μ-formula φ belongs to the class $\Sigma_0^\mu = \Pi_0^\mu$ if and only if it contains no fixed point operators. The class Σ_{n+1}^μ is defined to be the closure of $\Sigma_n^\mu \cup \Pi_n^\mu$ under the following rules:

1. If $\varphi, \psi \in \Sigma_{n+1}^\mu$, then $\varphi \wedge \psi, \varphi \vee \psi, \Box\varphi, \Diamond\varphi \in \Sigma_{n+1}^\mu$.
2. If $\varphi \in \Sigma_{n+1}^\mu$ and X occurs freely and positive in φ, then $\mu X.\varphi \in \Sigma_{n+1}^\mu$.
3. If $\varphi(X), \psi \in \Sigma_{n+1}^\mu$, then $\varphi(\psi) \in \Sigma_{n+1}^\mu$, provided that no free variable of ψ becomes bound by a fixed point operator in φ.

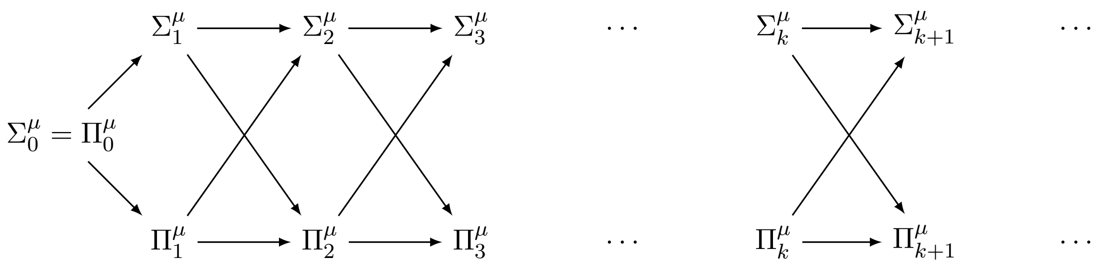
The class Π_{n+1}^μ is defined to be the closure of $\Sigma_n^\mu \cup \Pi_n^\mu$ under the following rules:

1. If $\varphi, \psi \in \Pi_{n+1}^\mu$, then $\varphi \wedge \psi, \varphi \vee \psi, \Box\varphi, \Diamond\varphi \in \Pi_{n+1}^\mu$.
2. If $\varphi \in \Pi_{n+1}^\mu$ and X occurs freely and positive in φ, then $\nu X.\varphi \in \Pi_{n+1}^\mu$.
3. If $\varphi(X), \psi \in \Pi_{n+1}^\mu$, then $\varphi(\psi) \in \Pi_{n+1}^\mu$, provided that no free variable of ψ becomes bound by a fixed point operator in φ.

Definition 2.4.1. The *alternation depth* of a formula φ is the least natural number n such that $\varphi \in \Sigma_{n+1}^\mu \cap \Pi_{n+1}^\mu$.

The formula *always eventually* belongs to $\Sigma_2^\mu \cap \Pi_2^\mu$ and has therefore alternation depth 1. The formula *infinitely often* belongs to $\Sigma_3^\mu \cap \Pi_3^\mu$, which implies that its alternation depth is 2. The fragment $\Sigma_2^\mu \cap \Pi_2^\mu$ is called the *alternation free fragment* of the modal mu-calculus. The *alternation depth hierarchy* consists of the classes Σ_n^μ and Π_n^μ for all $n \in \omega$ ordered by the subset relation. The picture below shows the alternation depth hierarchy, where the arrows represent the subset relation. For example $\Sigma_1^\mu \longrightarrow \Sigma_2^\mu$ encodes that $\Sigma_1^\mu \subseteq \Sigma_2^\mu$.

[1] This becomes more clear when one considers what is called fixed point approximations. By computing the fixed point approximation of the *always eventually* formula, one has to compute two independent fixed point approximations, the inner and the outer one. In the *infinitely often* formula however, one has to compute the inner approximations in each step of the outer approximations, resulting in much higher complexity. For an introduction to fixed point approximations we refer to [5].



The alternation depth hierarchy is sometimes also called the Niwiński hierarchy and was shown to be strict by Bradfield in 1996 [4].

Theorem 2.4.2 (Bradfield 1996)**.** For every natural number $n > 0$, there exists a formula $\varphi \in \Sigma_n^\mu$ which is not equivalent to any formula in Π_n^μ.

As mentioned in the introduction, much of the later work is focused on the first level of the alternation depth hierarchy.

Definition 2.4.3. The *first level of the alternation depth hierarchy* is defined to be $\Sigma_1^\mu \cup \Pi_1^\mu$.

Observe that by definition $\Sigma_1^\mu \cup \Pi_1^\mu$ consists only of formulas that have syntactic fixed point alternation depth of at most one, that is modal formulas as well as formulas that only contain least fixed point operators or only contain greatest fixed point operators. The first level of the alternation depth hierarchy is not identical with the alternation free fragment. The former is $\Sigma_1^\mu \cup \Pi_1^\mu$ and is strictly contained in the latter which is $\Sigma_2^\mu \cap \Pi_2^\mu$ [14].

Chapter 3

Tableaux, proof systems and model checking games

3.1 Introduction

The starting point of our proof theoretic investigation of the modal mu-calculus is the tableaux system **T**, which was introduced by Niwiński and Walukiewicz [16] in 1996.[1] A tableaux system is used to check whether formulas are satisfiable. It is called *sound*, if every formula that has a derivation in the system is satisfiable and *complete*, if the converse holds, namely if every formula which is satisfiable has a derivation. Derivations in a tableaux system are called tableaux. A tableau is a tree where every node is labelled by a set of formulas. In the tableaux system **T**, tableaux are in general finite branching trees which allow for infinite branches. These infinite branches are generated by the unfolding and regenerating of fixed point formulas. For example, for the greatest fixed point operator ν, there are two rules:

$$\frac{Z}{\nu Z.\varphi(Z)}\ (\nu) \qquad\qquad \frac{\varphi(Z)}{Z}\ (Z)$$

The rules are read bottom up. The left rule decomposes the fixed point formula $\nu Z.\varphi(Z)$ into the variable Z. The right rule then allows a regeneration of the body $\varphi(Z)$ of the fixed point formula. By applying rules for the Boolean connectives and the modal operators, we can then decompose the formula $\varphi(Z)$ until we reach a node labelled by Z higher up in the tableau. At this node we can regenerate the body again and so on. This leads to infinite branches. The tableaux system **T** is the foundation of the Gentzen-style proof systems presented in chapter 5 and the circular tableaux system in chapter 4. We therefore devote this chapter to properly introduce the system **T** and prove its soundness. Soundness and completeness of **T** was established by Niwiński and Walukiewicz [16] by using the close connection of tableaux and model

[1] The system that was introduced in [16] differs a bit in presentation from the tableaux system **T** in this chapter. The two systems are however equivalent.

checking games. We follow their approach in proving soundness. We first introduce model checking games in the next section 3.2 and afterwards the tableaux system **T** in section 3.3. The last section 3.4 consists of a detailed discussion of the soundness proof.

The presentation of this chapter closely follows the lecture notes of the course *Logic, Games and Automata* [2] as well as Niwiński's and Walukiewicz's original paper [16].

3.2 Model checking games

From now on and for the rest of the thesis we assume formulas to be in guarded normal form. In this section we introduce model checking games. These are infinitary two player games which are used to answer the model checking problem.

The model checking problem: Given a transition system T, a state s, a valuation V and a L_μ-formula φ, does $s \in [\![\varphi]\!]_V^T$ hold?

Recall the definition of a directed graph.

Definition 3.2.1. A *directed graph* is a tuple $\langle \mathcal{V}, E \rangle$ where $\mathcal{V}$ is a set of *vertices* and $E \subseteq \mathcal{V} \times \mathcal{V}$ is an ordered set of vertices. Given $u, v \in \mathcal{V}$ such that $(u, v) \in E$, we say that there is an *edge* from u to v, written $u \to v$.

Let φ be a L_μ-formula and fix a transition system $T = (S, \to, \rho)$, a state $s \in S$ and a valuation V. The *model checking game* $\mathcal{G}_V^T(s, \varphi)$ with respect to the system T, valuation V, state s and formula φ consists of two players:

- ▷ The *Verifier*, whose goal is to show that $T, V, s \models \varphi$.
- ▷ The *Refuter*, whose goal is to show that $T, V, s \not\models \varphi$.

The game is played on a directed graph which is called the *arena* of $\mathcal{G}_V^T(s, \varphi)$.

- ▷ Vertices of the arena are pairs (t, ψ) where $t \in S$ and $\psi \in Sub(\varphi)$;
- ▷ The existence of edges between vertices depends on the shape of the subformulas of the vertices.
  - Boolean subformulas: For each $t \in S$ and each subformula of φ of the form $\psi_1 \wedge \psi_2$ or $\psi_1 \vee \psi_2$ there are the following edges:

    $$(t, \psi_1 \wedge \psi_2) \longrightarrow (t, \psi_i) \quad (t, \psi_1 \vee \psi_2) \longrightarrow (t, \psi_i)$$

    for $i \in \{1, 2\}$.
  - Modal subformulas: For each $t \in S$ and each $u \in S$ such that $t \to u$ and each subformula of φ of the form $\Box\psi$ or $\Diamond\psi$ there are the following edges:

$$(t, \Box\psi) \longrightarrow (u, \psi) \quad (t, \Diamond\psi) \longrightarrow (u, \psi)$$

– Fixed point subformulas: For each $t \in S$ and each subformula of φ of the form $\sigma Z.\psi$ where $\sigma \in \{\mu, \nu\}$ there are the following edges:

$$(t, \sigma Z.\psi) \longrightarrow (t, Z) \quad (t, Z) \longrightarrow (t, \psi)$$

Given a vertex (t, ψ), there are no outgoing edges just if $\psi = P$ or $\psi = \neg P$ for $P \in Prop$ or if $\psi = Z$ or $\psi = \neg Z$ where $Z \in Var$ is a free variable in φ. Every vertex of the arena is labelled by either Verifier or Refuter. This labelling indicates to which player the vertex belongs. If a vertex (t, ψ) is labelled by Verifier, then it is Verifier's turn to play when the game is in position (t, ψ) and similarly if the vertex is labelled by Refuter, then it is Refuter's turn to play. Playing means choosing the next vertex: If the current position of the game is (t, ψ), labelled by Player $\in$ {Verifier, Refuter} and there are vertices $(u_1, \psi_1), ..., (u_k, \psi_k)$ such that for all $1 \leq i \leq k$ it holds that $(t, \psi) \longrightarrow (u_i, \psi_i)$, then Player plays by choosing to move the game to the next vertex (u_j, ψ_j) where $j \in \{1, ..., k\}$. It is only allowed to choose vertices which are connected by an edge from the current position. Such vertices are also called *successor vertices*. In case there is only one successor vertex, Player has to choose that one. In case there are no successor vertices, the game ends. Obviously, it is only important to know whose turn it is in case the current vertex has out-degree larger than 1. For conciseness, we only assign players to vertices where the out-degree is possibly larger than 1 and assume that every other vertex with out-degree ≤ 1 is labelled by some player (which one does not matter). The tabular below indicates which (relevant) vertices belong to which player:

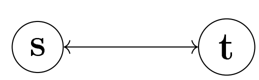
Verifier	Refuter
$(t, \psi_1 \vee \psi_2)$	$(t, \psi_1 \wedge \psi_2)$
$(t, \Diamond\psi)$	$(t, \Box\psi)$

Definition 3.2.2. A *play* of $\mathcal{G}_V^T(s, \varphi)$ is a sequence of vertices $(s_0, \varphi_0), (s_1, \varphi_1), (s_2, \varphi_2), ...$ such that $s_0 = s$, $\varphi_0 = \varphi$ and the following two conditions hold for all $i \in \omega$:

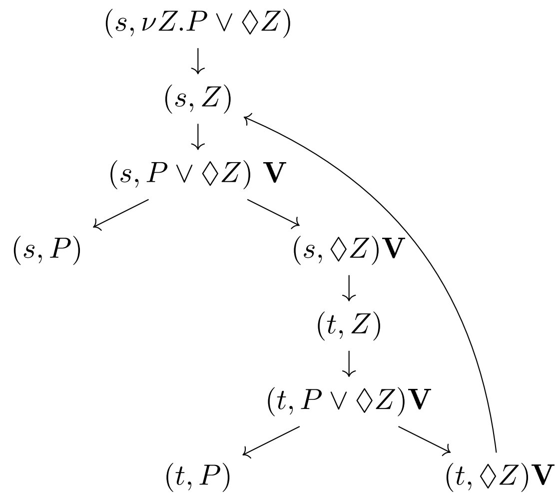
1. If (s_i, φ_i) has out-going edges, then $(s_i, \varphi_i) \longrightarrow (s_{i+1}, \varphi_{i+1})$. Otherwise the play ends.
2. If (s_i, φ_i) is labelled by Verifier (respectively Refuter), then Verifier (respectively Refuter) chooses (s_{i+1}, φ_{i+1}).

Let us consider an example.

Example 3.2.3. Let $\varphi = \nu Z.P \vee \Diamond Z$ and consider the following transition system T, consisting of two states s and t such that $s \rightarrow t$ and $t \rightarrow s$ where $\rho(P) = \{s\}$:



The arena of the model model checking game $\mathcal{G}_\emptyset^T(s, \varphi)$ is depicted below.



Observe that every relevant vertex of the arena is labelled by Verifier, abbreviated by **V**. When a play reaches the node $(s, P \vee \Diamond Z)$, Verifier gets to choose whether to move left or right. In case he moves left, the game ends as (s, P) is a dead end. If he moves right, the game continues. There is exactly one possible infinite play in this arena, namely when Verifier decides at each relevant node with out-degree larger than one to always go right.

Next, we formulate winning conditions for both Verifier and Refuter for a given play, for which we need the following concept.

Definition 3.2.4. Let φ be a L_μ-formula and let $\sigma_1 X_1.\psi_1$ and $\sigma_2 X_2.\psi_2$ be two subformulas of φ. The variable X_1 *subsumes* X_2 if and only if $\sigma_2 X_2.\psi_2 \in Sub(\sigma_1 X_1.\psi_1)$.

As an example, in the formula $(\mu Z.\Diamond(Z \vee \nu Y.\Box(Q \wedge Y))) \vee \nu X.\Box X$ the variable Z subsumes Y while the variable X neither subsumes Z or Y nor Z or Y subsume X. The following proposition is a standard result.

Proposition 3.2.5. If $(s_0, \varphi_0), (s_1, \varphi_1), \ldots (s_n, \varphi_n), \ldots$ is an infinite play in the model checking game $\mathcal{G}_V^T(s_0, \varphi_0)$, then there is a unique variable X such that

1. X occurs infinitely often in the play and

2. if Y also occurs infinitely often, then X subsumes Y.

The variable X occurring infinitely often in a play means that there are infinitely many vertices in the play whose second component is the variable X.

Definition 3.2.6. Let $\mathcal{G}_V^T(s_0, \varphi_0)$ be the model checking game for some transition system T, state s_0, valuation V and formula φ_0.

1. Verifier *wins* a play if

   (a) the play $(s_0, \varphi_0), ...(s_n, \varphi_n)$ is finite and
      i. $\varphi_n = P$ and $s_n \in \rho(P)$ or $\varphi_n = \neg P$ and $s_n \notin \rho(P)$
      ii. $\varphi_n = Z$ for Z free in φ_0 and $s_n \in V(Z)$ or $\varphi_n = \neg Z$ for Z free in φ_0 and $s_n \notin V(Z)$
      iii. $\varphi_n = \Box\psi$ and $\{t \in S | s_n \to t\} = \emptyset$
   (b) the play is infinite and the unique variable X that occurs infinitely often in the play and subsumes all other infinitely often occurring variables is a ν-variable.

2. Refuter *wins* a play if
   (a) the play $(s_0, \varphi_0), ...(s_n, \varphi_n)$ is finite and
      i. $\varphi_n = P$ and $s_n \notin \rho(P)$ or $\varphi_n = \neg P$ and $s_n \in \rho(P)$
      ii. $\varphi_n = Z$ for Z free in φ_0 and $s_n \notin V(Z)$ or $\varphi_n = \neg Z$ for Z free in φ_0 and $s_n \in V(Z)$
      iii. $\varphi_n = \Diamond\psi$ and $\{t \in S | s_n \to t\} = \emptyset$
   (b) or the play is infinite and the unique variable X that occurs infinitely often in the play and subsumes all other infinitely often occurring variables is a μ-variable.

Notice that every play is won by exactly one player. If the play is finite, then the last position (s_n, φ_n) is such that φ_n is either a literal or a (possibly negated) free variable or a modal formula where one of the players cannot extend the play. All six cases correspond to either Verifier or Refuter winning. If the play is infinite, then by proposition 3.2.5 there exists a unique variable that occurs infinitely often and subsumes every other variable that occurs infinitely often. Thus if this unique variable is a ν-variable, then Verifier wins and otherwise Refuter wins. Having defined winning conditions on specific plays, let us now define strategies. Intuitively, a strategy for a player is a set of rules that tells the player how to play in specific positions. A strategy is called memoryless, if the rules only depend on the current position of the game and not on previous moves and positions. Formally, memoryless strategies are defined as follows:

Definition 3.2.7. Let $\mathcal{G}_V^T(s_0, \varphi_0)$ be a model checking game and let Player $\in$ {Verifier, Refuter}. A *memoryless strategy* for Player is a function *Str* that maps every vertex (s, φ) of $\mathcal{G}_V^T(s_0, \varphi_0)$ labelled by Player to one of its successor vertices or to a distinguished token $\bot$ in case there are no successor vertices.

Given a strategy *Str* for Player $\in$ {Verifier,Refuter}, we say that Player *uses Str* if whenever the play is in a position (s, φ) labelled by Player, he chooses to move to the vertex $Str((s, \varphi))$, that is, he plays according to the strategy.

Definition 3.2.8. A memoryless strategy *Str* for Player $\in$ {Verifier, Refuter} is *winning*, if Player wins every play in which he uses *Str*.

In case some player has a winning strategy, it follows from our previous observation that the other player does not have a winning strategy. The following theorem shows that winning strategies always exist.

Theorem 3.2.9. Given a model checking game $\mathcal{G}_V^T(s_0, \varphi_0)$, exactly one of Verifier and Refuter has a memoryless winning strategy.

The theorem is a corollary of Martin's result in 1975 that every Borel game is determined (which means that exactly one of the two players has a winning strategy) and the fact that model checking games as defined here are Borel games. We skip the proof of this result and refer the reader to Martin's original paper, namely [13].

Example 3.2.10. Recall the model checking game from example 3.2.3. Verifier wins a play in this game, if the play is finite and ends in node (s, P) or if the play is infinite (notice that there is only a single infinite play), because the unique variable occurring in the infinite play is a ν-variable. Therefore, the strategy that tells Verifier to go left at node $(s, P \vee \Diamond Z)$ is winning. Moreover the strategy defined by

- ▷ At node $(s, P \vee \Diamond Z)$ go to $(s, \Diamond Z)$
- ▷ At node $(t, P \vee \Diamond Z)$ go to $(t, \Diamond Z)$

is a winning strategy as well. Observe that both strategies are memoryless. Notice that Refuter can win a play, namely if Verifier chooses to go to the left at node $(t, P \vee \Diamond Z)$ but he does not have a winning strategy.

We finish this section by stating the Fundamental Semantic Theorem, which establishes the connection between truth of a formula in a given state of a transition and the existence of memoryless winning strategies for Verifier in the associated model checking game.

Theorem 3.2.11 (Fundamental Semantic Theorem; Streett and Emerson 1989)**.** Let $T = (V, \rightarrow, \rho)$ be a transition system, $s \in S$ a state, V a valuation and φ a L_μ-formula.

$$T, V, s \models \varphi \Leftrightarrow \text{Verifier has a memoryless winning strategy for } \mathcal{G}_V^T(s, \varphi)$$

For the proof we refer the reader to Streett and Emerson's original paper [17].

3.3 The tableaux system T

This section introduces the tableaux system **T**. We already assumed formulas to be in guarded normal form in the last section. From now on we also assume that every formula is closed. The tableaux system **T** operates on sequents.

Definition 3.3.1. A *sequent* is a finite set of L_μ-formulas.

Sequents are denoted by the capital Greek letters $\Gamma, \Delta, \Sigma, \Pi, \Omega, \Phi$ and Θ, where we add sub- or superscripts when needed. Given a sequent $\Gamma = \{\varphi_1, ..., \varphi_n\}$, its *interpretation* $\mathcal{I}(\Gamma)$ is defined to be the conjunction over all formulas that belong to Γ:

$$\mathcal{I}(\Gamma) := \bigwedge \Gamma = \varphi_1 \wedge (\varphi_2 \wedge (... \wedge \varphi_n)...)$$

We call a sequent Γ *satisfiable*, if $\mathcal{I}(\Gamma)$ is satisfiable, that is if there exists a transition system $T = (S, \rightarrow, \rho)$ and a state $s \in S$, such that every formula $\varphi \in \Gamma$ is true in that state. Notice that we do not need to consider valuations as every formula is assumed to be closed. A set $U \subseteq Lit$ of literals is called *inconsistent*, if $P, \neg P \in U$ for some $P \in Prop$ and *consistent* otherwise. Given a sequent Γ, let $\Box\Gamma := \{\Box\varphi | \varphi \in \Gamma\}$ and let $\Diamond\Gamma := \{\Diamond\varphi | \varphi \in \Gamma\}$. Moreover, writing Γ, φ is short for $\Gamma \cup \{\varphi\}$ and Γ, Δ for $\Gamma \cup \Delta$.

Table 3.1: The tableaux system **T**

$\dfrac{\Gamma, \varphi_0, \varphi_1}{\Gamma, \varphi_0 \wedge \varphi_1}\ (\wedge)$	$\dfrac{\Gamma, \varphi_0}{\Gamma, \varphi_0 \vee \varphi_1}\ (\vee)_0$	$\dfrac{\Gamma, \varphi_1}{\Gamma, \varphi_0 \vee \varphi_1}\ (\vee)_1$
$\dfrac{\Gamma, Z}{\Gamma, \mu Z.\varphi(Z)}\ (\mu)$	$\dfrac{\Gamma, Z}{\Gamma, \nu Z.\varphi(Z)}\ (\nu)$	$\dfrac{\Gamma, \varphi(Z)}{\Gamma, Z}\ (Z)$
	$\dfrac{\Gamma, \varphi_1 \quad ... \quad \Gamma, \varphi_n}{\Box\Gamma, \Diamond\varphi_1, ..., \Diamond\varphi_n, \Theta}\ (mod)$	$(\Theta \subseteq Lit$ consistent$)$

Definition 3.3.2. The tableaux system **T** consists of the following inference rules:

1. the Boolean rules $(\wedge)$, $(\vee)_0$ and $(\vee)_1$
2. the modality rule (mod)
3. the fixed point rules (μ), (ν) and (Z)

and is depicted in table 3.1.

In the rule (Z) it is assumed that the variable Z identifies the formula $\varphi(Z)$ and that this identification is unique.[2] In the rule (mod) the set Γ is allowed to be empty, but at least one diamond formula is required to apply the rule. Also notice that the side sequent of literals Θ is required to be consistent. Thus if Θ is inconsistent, the rule cannot be applied. Notice that in such a situation no rule can be applied any more. The notion of a pre-tableau and a tableau depends on the notion of a labelled tree. Recall that a *partial order* is a reflexive,

[2]In the sense that in a given sequent every variable only identifies one formula. Thus $\sigma Z.\varphi(Z) \neq \sigma' Y.\varphi'(Y)$ implies that $X \neq Y$.

transitive and antisymmetric binary relation and a *linear order* is a binary relation which is transitive, antisymmetric and connex.

Definition 3.3.3. A *tree* is a tuple $\langle V, \rightarrow \rangle$ where V is a set and $\rightarrow$ is a partial order on V such that:

1. There exists an element $x \in V$, which is called the *root*, such that for all $y \in V : x \rightarrow y$
2. $\langle \{y \in V \mid y \rightarrow y'\}, \rightarrow \rangle$ is linearly ordered for all $y' \in V$

The following notation is used:

- ▷ Each $y \in V$ is called a *node*.
- ▷ If $y \in V$ such that there exists no $x \in V$ with $y \rightarrow x$ and $y \neq x$, then y is called a *leaf*.
- ▷ A node x is a *child* of a *parent* node y if $y \neq x$, $y \rightarrow x$ and for all z such that $z \neq y$ and $z \neq x$, if $y \rightarrow z$, then $z \not\rightarrow x$.

A *labelled tree* (with respect to a set A) is a triple $t = (V, \rightarrow, \lambda)$ where $(V, \rightarrow)$ is a tree and $\lambda : V \longrightarrow A$ is a *labelling function* that assigns each node of t an element in A.

Definition 3.3.4. A *pre-tableau* for a sequent Γ is a labelled tree $t = (V, \rightarrow, \lambda)$ with respect to $\mathcal{P}(\Gamma)$ generated by the tableaux rules of **T** such that

1. $\lambda(r_t) = \Gamma$ where r_t denotes the root of t and
2. every leaf of t is labelled by a sequent of the form $\Box\Delta, \Diamond\Pi, \Theta$ where $\Theta \subseteq Lit$ and either
   - $\Pi = \emptyset$ or
   - Θ is inconsistent

Pre-tableaux are read bottom-up. Every rule in **T** with the exception of (mod) operates on a single formula. In the case of $(\wedge)$, this formula is a conjunction and the rule decomposes it into the two conjuncts. In the case of $(\vee)_0$ this formula is a disjunction and the rule decomposes it into the left disjunct and so on. We call these relevant formulas in the conclusion and premise of a rule the *distinguished formulas of the rule*. In the (mod)-rule we consider every formula as distinguished. For each rule the distinguished formula(s) in the lower sequent is (are) called *principal* and the distinguished formula(s) in the upper sequent is (are) called *residual*. The other formulas are called *side-formulas*. For example in the rule

$$\frac{\Gamma, \varphi_0, \varphi_1}{\Gamma, \varphi_0 \wedge \varphi_1} \ (\wedge)$$

the formula $\varphi_0 \wedge \varphi_1$ is the principal formula and φ_0 and φ_1 are the residual formulas, while the formulas in Γ are side-formulas.

The notion of pre-tableau fixes the kind of infinite trees which are considered. Observe that a pre-tableau is a finite branching tree: Branching only occurs when a (mod)-rule is applied and since there are only finitely many premises for each instance of (mod), the branching is finite. The fixed point regeneration rule (Z) allows for infinite branches, as we can regenerate the formula which is identified by the variable Z and then continue to apply other rules to decompose that formula until we are back with the formula Z, which then can be regenerated again and so on.

Example 3.3.5. Consider the formula $\varphi = \nu Z.\Diamond(Z \vee (P \wedge \neg P))$. The following is a pre-tableau for φ:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\text{etc.}}{Z}}{Z \vee (P \wedge \neg P)}\,(\vee)_0}{\Diamond(Z \vee (P \wedge \neg P))}\,(mod)}{Z}\,(Z)}{Z \vee (P \wedge \neg P)}\,(\vee)_0}{\Diamond(Z \vee (P \wedge \neg P))}\,(mod)}{Z}\,(Z)}{\nu Z.\Diamond(Z \vee (P \wedge \neg P))}\,(\nu)$$

The uppermost label etc. denotes that we keep extending the branch by choosing the variable Z at the disjunction $Z \vee (P \wedge \neg P)$. Therefore the pre-tableau is infinite. Notice that there are several different pre-tableaux for φ. For example, the following is a finite pre-tableau:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P, \neg P}{P \wedge \neg P}\,(\wedge)}{Z \vee (P \wedge \neg P)}\,(\vee)_1}{\Diamond(Z \vee (P \wedge \neg P))}\,(mod)}{Z}\,(Z)}{\nu Z.\Diamond(Z \vee (P \wedge \neg P))}\,(\nu)$$

By following the variable Z for finitely many steps and then ending the branch by going through $P \wedge \neg P$ one obtains finite pre-tableaux for φ which are different to the one shown above. Indeed it is easy to see that there are infinitely many finite pre-tableaux and exactly one infinite pre-tableau for φ.

The conditions imposed on a pre-tableau in order to be a tableau depend on the notion of a trace trough a path. We start by defining the notion of a path through a pre-tableau.

Definition 3.3.6. Let $t = (V, \to, \lambda)$ be a pre-tableau with root r_t. A *path* through t is a (possibly infinite) sequence of nodes $\mathbb{P} = \mathbb{P}(0)\mathbb{P}(1)\mathbb{P}(2)...$ in V such that $\mathbb{P}(0) = r_t$ and for all $i \in \omega$ for which $\mathbb{P}(i)$ exists it holds that:

1. If $\mathbb{P}(i)$ is not a leaf, then $\mathbb{P}(i) \to \mathbb{P}(i+1)$.
2. If $\mathbb{P}(i)$ is a leaf, then $\mathbb{P} = \mathbb{P}(0)\mathbb{P}(1)...\mathbb{P}(i)$ (that is, the path ends at $\mathbb{P}(i)$).

Notice that finite paths have to end in a leaf. Given that the last node in a finite path is $\mathbb{P}(n)$, we say that the *length* of $\mathbb{P}$ is $n+1$ (written $lth(\mathbb{P}) = n+1$). The following proposition is a well-known result about the connection of infinite paths to the (mod)-rule; the proof is standard and omitted.

Proposition 3.3.7. Every infinite path in a pre-tableau passes through a (mod)-rule infinitely often.

Given a sequence $a = a_0, a_1, a_2, ...$ an *initial segment of a* is a finite sequence $b_0, ..., b_k$ such that for all $0 \leq i \leq k$ it holds that $b_i = a_i$.

Definition 3.3.8. Let $t = (V, \to, \lambda)$ be a pre-tableau for some sequent Γ and let $\mathbb{P}$ be a path through t. A finite sequence of formulas $\varphi_0, \varphi_1, ..., \varphi_n$ is a *finite trace* through $\mathbb{P}$, if

1. $\varphi_i \in \lambda(\mathbb{P}(i))$ for all $0 \leq i \leq n$ and
2. if φ_i is not principal in the rule from $\mathbb{P}(i)$ to $\mathbb{P}(i+1)$, then $\varphi_i = \varphi_{i+1}$ and otherwise φ_{i+1} is (one of) the residual subformula(s) of φ_i.

An infinite sequence of formulas $\varphi_0, \varphi_1, ...\varphi_n, ...$ is an *infinite trace*, if every initial segment of the sequence is a finite trace.

Lemma 3.3.9. For every infinite trace there exists a unique variable that occurs infinitely often and subsumes every other variable that occurs infinitely often.

The proof is standard and we omit it. An infinite trace is called a *μ-trace*, if the unique variable identified by the lemma is a μ-variable and it is called a *ν-trace*, if this variable is a ν-variable.

Definition 3.3.10. A *tableau* for Γ is a pre-tableau $t = (V, \to, \lambda)$ for Γ such that

1. every leaf of t is labelled by a sequent of the form $\Box\Delta, \Theta$ where $\Theta \subseteq Lit$ is consistent
2. every infinite trace is a ν-trace.

Recall the two pre-tableaux displayed in example 3.3.5. The first pre-tableau is a tableau, as there are no leafs and the only infinite trace is a ν-trace. The second pre-tableau however is not a tableau, as there is a leaf which is labelled by inconsistent literals. We draw two important conclusions from that example. First, pre-tableaux (and also tableaux) are not unique. A sequent can have several and even infinitely many different pre-tableaux. Second, a sequent having a tableau does not imply that every pre-tableau for that sequent is a tableau. Indeed in example 3.3.5, the formula φ has infinitely many pre-tableaux of which only a single one is a tableau.

3.4 Soundness of T

This section establishes the soundness of the tableaux system **T**. That is, we show that if a sequent Γ has a tableau, then Γ is satisfiable. As the proof technique for obtaining soundness is used in later chapters, we provide a detailed proof in this section. Soundness and completeness of **T** was proven by Niwiński and Walukiewicz in [16]. Notice that the tableaux system used by Niwiński and Walukiewicz differs in presentation. However, their approach can easily be adjusted for our system. While we give a detailed soundness proof, we only state the completeness result and refer for its proof to Niwiński and Walukiewicz original paper [16].

Theorem 3.4.1 (Completeness of **T**, Niwiński and Walukiewicz 1996)**.** If a sequent Γ is satisfiable, then it has a tableau.

In order to prove that **T** is sound, we use the model checking games introduced in section 3.2. Given a tableau for Γ, we show how to build a transition system $T = (S, \rightarrow, \rho)$ and a state $s \in S$ and then provide a memoryless winning strategy for Verifier in the model checking game $\mathcal{G}^T_\emptyset(s, \varphi)$ for any $\varphi \in \Gamma$. The winning strategy for Verifier is thereby based on the close connection between tableaux and model checking games. We show that if Verifier plays according to the provided strategy, every play corresponds to a trace in the tableau, from which we deduce that Verifier wins every play. The Fundamental Semantic Theorem then implies that every φ holds at state s in the system T which in turn implies that $\bigwedge \Gamma$ holds at state s and so that Γ is satisfiable. In order to show that Verifier wins finite plays, we require the following lemma.

Lemma 3.4.2. Let $t = (V, \rightarrow, \lambda)$ be a tableau and let $v \in V$. For all $P \in Prop$ it holds that $\{P, \neg P\} \not\subseteq \lambda(v)$.

The proof of the lemma is based on the observation that the only rule which applies weakening is (mod) and (mod) can only be applied when the side-sequent of literals is consistent. That is, if a node was labelled by inconsistent literals, then neither at this node nor at any later node the (mod)-rule could be applied, which implies that every path through that node is finite and leads to a leaf labelled by inconsistent literals. Therefore no tableau can have a node labelled by inconsistent literals.

Theorem 3.4.3 (Soundness of **T**, Niwiński and Walukiewicz 1996)**.** If a sequent Γ has a tableau, then Γ is satisfiable.

Proof. Suppose Γ is a sequent and $t = (V, \rightarrow_t, \lambda)$ is a tableau for Γ with root r_t. We define a transition system $T = (S, \rightarrow_T, \rho)$ and a map $\tau : V \longrightarrow S$ using the tableau t, such that the following conditions hold:

- ▷ $\tau(r_t) = s_0$ for $s_0 \in S$.
- ▷ Suppose $u \rightarrow_t v$. If the rule applied at u is (mod), then $\tau(u) \neq \tau(v)$ and $\tau(u) \rightarrow_T \tau(v)$, otherwise $\tau(u) = \tau(v)$.

▷ $s \in \rho(P)$ if and only if there exists $v \in V$ such that $\tau(v) = s$ and $P \in \lambda(v)$.

Observe that T is a well-defined transition system: It consists of a non-empty set of states S (non-empty because we stipulated that $s_0 \in S$) and $\rightarrow_T$ is a binary relation defined on S. Moreover $\rho : Prop \longrightarrow \mathcal{P}(S)$ is clearly well-defined. Notice that T is itself a tree where each node corresponds to (several) nodes in t and there is a transition between two nodes s_1 and s_2 if and only if the uppermost corresponding vertex u of s_1 in t has an edge to the lower-most corresponding vertex v of s_2 and moreover the rule applied at u is (mod). The root of the tree T is the distinguished state s_0. We claim that $T, \emptyset, s_0 \models \varphi$ for all $\varphi \in \Gamma$. Let φ be an arbitrary formula of Γ and consider the model checking game $\mathcal{G}_{\emptyset}^{T}(s_0, \varphi)$. We show that Verifier has a memoryless winning strategy and in particular that every play corresponds to a trace through t. Notice that every play starts in position (s_0, φ_0) where $\varphi_0 = \varphi$. We only consider those traces in t that start in φ_0. We say that the initial segment (s_0, φ_0) of every play corresponds to the initial segment φ_0 of every (relevant) trace. Now suppose we have an initial segment of a play

$$(s_0, \varphi_0), (s_1, \varphi_1), ...(s_n, \varphi_n)$$

which corresponds to the initial segment of a trace

$$\varphi_0, ..., \varphi_0, \varphi_1, ..., \varphi_1, ..., \varphi_n$$

such that $\varphi_n \notin Lit$. We show how the play and the trace can be extended:

Case 1: It is Verifier's move. This implies that φ_n is either $\psi_0 \vee \psi_1$ or $\Diamond\psi$.

▷ Suppose $\varphi_n = \psi_0 \vee \psi_1$. Therefore Verifier can choose to move to (s_n, ψ_0) or to (s_n, ψ_1). Suppose the lowermost associated node (by τ) to s_n is v. Since t is a tableau, there exists a node u reachable from v at which the rule $(\vee)_i$ is applied to φ_n for $i \in \{0, 1\}$ decomposing $\psi_0 \vee \psi_1$ into ψ_i. Suppose there are $k - 1 \geq 0$ steps between v and u. Then we extend the trace to

$$\varphi_0, ..., \varphi_0, \varphi_1, ..., \varphi_1, ..., \underbrace{\varphi_n, ..., \varphi_n}_{k-times}, \varphi_{n+1}$$

where $\varphi_{n+1} = \psi_i$. We let Verifier extend the play to

$$(s_0, \varphi_0), (s_1, \varphi_1), ...(s_n, \varphi_n), (s_{n+1}, \varphi_{n+1})$$

where $s_{n+1} = s_n$ and $\varphi_{n+1} = \psi_i$.

▷ Suppose $\varphi_n = \Diamond\psi$ and the lowermost associated node to s_n is v. Since t is a tableau there exists a node u reachable from v which is also labelled by $\Diamond\psi$ and at which the rule applied is (mod) splitting the branch into l branches where one immediate successor of u, say w, is labelled by ψ. Suppose there are $k - 1 \geq 0$ vertices between v and u. Then we can extend the trace to

$$\varphi_0, ..., \varphi_0, \varphi_1, ..., \varphi_1, ..., \underbrace{\varphi_n, ..., \varphi_n}_{k-times}, \varphi_{n+1}$$

where $\varphi_{n+1} = \psi$. Notice that by construction of T each vertex between v and u is associated to s_n. Moreover $s_n \neq \tau(w)$ and $s_n \rightarrow_T \tau(w)$. Thus in the game there exists a position $(\tau(w), \psi)$ which Verifier can choose. We let Verifier extend the play to

$$(s_0, \varphi_0), (s_1, \varphi_1), ...(s_n, \varphi_n), (s_{n+1}, \varphi_{n+1})$$

where $s_{n+1} = w$ and $\varphi_{n+1} = \psi$.

Case 2: It is Refuter's move. This implies that φ_n is either $\psi_0 \wedge \psi_1$ or $\Box\psi$. We show that no matter what choice Refuter takes to extend the play, we can extend the trace accordingly.

- ▷ Suppose $\varphi_n = \psi_0 \wedge \psi_1$. Thus Refuter can choose to extend the play by moving to (s_n, ψ_0) or to (s_n, ψ_1). Suppose he chooses to move to (s_n, ψ_i) for $i \in \{0, 1\}$. Let v be the lowermost associated node of s_n. By assumption v is labelled by $\psi_0 \wedge \psi_1$. Since t is a tableau there exists a node u reachable from v in $k - 1 \geq 0$ steps which is labelled by $\psi_0 \wedge \psi_1$ and the rule applied at u is $(\wedge)$ such that the successor of u, say w, is labelled by ψ_0, ψ_1. Notice that each node between v and u as well as w are associated to s_n. Therefore we extend the trace to

$$\varphi_0, ..., \varphi_0, \varphi_1, ..., \varphi_1, ..., \underbrace{\varphi_n, ..., \varphi_n}_{k-times}, \varphi_{n+1}$$

where $\varphi_{n+1} = \psi_i$. Notice that since both ψ_0 and ψ_1 are present at w, whatever choice Refuter takes we can choose the same formula to extend the trace. By construction the extended trace and play are still corresponding.

- ▷ Suppose $\varphi_n = \Box\psi$ and let v be the lowermost vertex that is associated to s_n. So s_n is labelled by $\Box\psi$. We assume that Refuter can extend the play (otherwise Verifier wins). Therefore there exists a vertex u reachable from v in $k-1$ steps such that the rule applied at u is (mod) and there are $l \geq 1$ children of u, say $u_1, ..., u_l$, which are all labelled by ψ. Notice that each of the vertices between v and u is associated by τ to s_n. Moreover u is associated to s_n as well and s_n has exactly l successors, namely $\tau(u_1), ..., \tau(u_l)$. Suppose Refuter chooses to move to $(\tau(u_i), \psi)$. Then the trace can be extended to

$$\varphi_0, ..., \varphi_0, \varphi_1, ..., \varphi_1, ..., \underbrace{\varphi_n, ..., \varphi_n}_{k-times}, \varphi_{n+1}$$

where $\varphi_{n+1} = \psi$ labels the node u_i.

Case 3: It is a neutral move. This implies that φ_n is either $\mu Z.\psi(Z)$, or $\nu Z.\psi(Z)$ or Z. It follows immediately that in all three cases both the play and the trace can be extended. We omit the details.

In case 1 we provided a strategy for Verifier by following the current trace through t. Notice that the strategy provided is memoryless. Furthermore, we have shown that if Verifier plays that strategy, then every play corresponds to some trace. This does not depend on how Refuter plays (indeed different choices by Refuter only impact which trace we follow). It remains to show that the strategy for Verifier is winning. For that, first suppose that we have a finite play $(s_0, \varphi_0), (s_1, \varphi_1), ...(s_n, \varphi_n)$ where Verifier uses the described strategy. The formula φ_n is thus either a literal or it is a boxed formula, where Refuter could not extend the play or it is a diamond formula where Verifier could not extend the play. Let $\varphi_0, ..., \varphi_0, ..., \varphi_n$ be the corresponding trace to the play and v the associated vertex of s_n. We distinguish three cases:

1. Suppose $\varphi_n \in Lit$. First suppose $\varphi_n = P$. Then $P \in \lambda(v)$ and since $\tau(v) = s_n$, we have by definition that $s_n \in \rho(P)$. Second suppose $\varphi_n = \neg P$. Again this implies that $\neg P \in \lambda(v)$ and so by lemma 3.4.2 it follows that $P \notin \lambda(v)$. Suppose that $s_n \in \rho(P)$. Then there must exist a vertex u after v such that $\tau(u) = s_n$ and $P \in \lambda(u)$. But $\tau(u) = s_n$ implies that there is no application of (mod) between v and u, which implies that $\neg P \in \lambda(u)$, thus contradicting lemma 3.4.2. Hence $s_n \notin \rho(P)$. In both cases Verifier wins.

2. Suppose $\varphi_n = \Box\psi$ and Refuter could not extend the play. This directly implies that Verifier wins.

3. Suppose $\varphi_n = \Diamond\psi$ and Verifier could not extend the play. If this was the case, the corresponding trace ends in the formula $\Diamond\psi$. Now suppose it is possible to extend the trace to ψ. In that case Verifier could have extended the play according to the strategy, as extending the trace implies that there exists a successor node of s_n in the transition system. Hence the trace cannot be extended, which implies that there is some leaf labelled by $\Diamond\psi$, contradicting our assumption that t is a tableau. Therefore this case cannot occur.

Hence, Verifier wins every finite play. Now suppose we have an infinite play $(s_0, \varphi_0), (s_1, \varphi_1), ...$ corresponding to the infinite trace $\varphi_0, ...\varphi_0, \varphi_1, ...\varphi_1, ...$ in t. Since t is a tableau every infinite trace is a ν-trace, which means that the variable that occurs infinitely often in the trace and subsumes all other infinitely often occurring variables is a ν-variable. This directly implies that the variable that occurs infinitely often in the play and subsumes all other infinitely often occurring variables is a ν-variable as well. Therefore Verifier wins every infinite play. Together we conclude that Verifier wins every play in $\mathcal{G}^T_\emptyset(s_0, \varphi)$, if he plays according to the strategy. Thus there is a memoryless winning strategy for Verifier. The Fundamental Semantic Theorem implies that the formula φ is true at state s_0 of the transition system T. As φ was an arbitrary formula in Γ, we conclude that $T, \emptyset, s_0 \models \bigwedge \Gamma$ and so that Γ is satisfiable. $\Box$

Chapter 4

Finite model property

4.1 Introduction

This chapter provides our first contribution towards investigating the mathematical theory of the first level of the alternation hierarchy. We establish that the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$ enjoys the finite model property. This property states that if a sequent Γ is satisfiable, then it is satisfiable in a finite model. Our proof strategy is based on the notion of a regular tableau.[1]

Definition 4.1.1. A tree is called *regular* if it contains only finitely many subtrees (up to isomorphisms) or equivalently, if the tree is the unfolding of a finite tree.

We call a tableau *regular*, if the underlying tree of a tableau is regular. For instance, the infinite tableau in example 3.3.5 is regular. It has only four distinct subtrees up to isomorphisms. We can also view the tableau as the unfolding of the following finite tree,

$$\cfrac{\cfrac{\cfrac{\cfrac{Z}{Z \vee (P \wedge \neg P)}\,(\vee)_0}{\Diamond(Z \vee (P \wedge \neg P))}\,(mod)}{Z}\,(Z)}{\nu Z.\Diamond(Z \vee (P \wedge \neg P))}\,(\nu)$$

where we identify the two nodes labelled by Z. To obtain the original infinite tableau, we unfold the finite tree over the two identified nodes. Regular tableaux are extraordinary well-behaved. Despite their infinite size they only carry a finite amount of information which is memorized in the finite structure that unfolds into the tableau. Suppose we have a regular tableau for some sequent Γ. By following the construction of the model in the soundness proof of **T**, we build a model for Γ, whose underlying frame is a tree. As the tableau is regular,

[1]Notice that the standard method to establish the finite model property for modal logics is the filtration technique. Unfortunately, filtration does not work for the modal mu-calculus, see [5].

the underlying frame is indeed a regular tree. Therefore we can prune the model at the leafs of the finite tree that unravels into the model and add loops from the leaves to earlier states to obtain an finite model for Γ. Therefore, the question whether the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$ has the finite model property reduces to the question whether every satisfiable sequent in that fragment has a regular tableau. We give a positive answer to that question by introducing a circular tableaux system for $\Sigma_1^\mu \cup \Pi_1^\mu$. A circular tableau is thereby a finite tree generated by the tableaux rules of $\mathbf{T}$, such that some leafs contain loops back to earlier nodes in the tree. We show that such circular tableaux unfold into infinitary regular tableaux. That is, the circular tableaux are exactly the finite trees in the definition of regular trees. By proving soundness and completeness of the circular tableaux system we establish that every satisfiable sequent in $\Sigma_1^\mu \cup \Pi_1^\mu$ has a regular tableau. The finite model property follows directly from the constructed model in the soundness proof.

The construction of the circular tableaux system depends on the lack of fixed point alternation in the formulas of the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$. As soon as we leave this fragment, our system is no longer sound. The finite model property was established for the whole modal mu-calculus by Emerson and Streett [17] in 1989. Their approach aims at showing that every satisfiable formula has a regular tree model. As Emerson and Streett work in the whole modal mu-calculus with arbitrary fixed point alternation, they use more sophisticated methods from advanced automata theory to find regular structures.

Before we define the above mentioned circular tableaux system, we briefly discuss why the fragment Σ_1^μ enjoys the finite model property in section 4.2. This discussion sheds some light onto the technical details of the definition of circular tableaux in section 4.3. Apart from the definition of the system, section 4.3 also consists of a brief discussion why the restriction to the first level of the alternation hierarchy is relevant for the soundness of the system. The last section 4.4 establishes the soundness and completeness of the circular tableaux system and thereby the finite model property of $\Sigma_1^\mu \cup \Pi_1^\mu$.

4.2 Finite model property for Σ_1^μ

This section establishes that the fragment Σ_1^μ enjoys the finite model property. Σ_1^μ is the class of formulas consisting of modal formulas and formulas containing only least fixed point operators. This implies that every infinite trace that starts in Σ_1^μ-formula is by definition a μ-trace. Recall that in a tableau every infinite trace is a ν-trace. Therefore, every trace in a tableau for a sequent of Σ_1^μ-formulas is finite. We prove that this implies that every such tableau is finite. For that we require König's Lemma. A labelled tree $t = (V, \rightarrow, \lambda)$ is infinite if and only if the set V is infinite.

Theorem 4.2.1 (König's Lemma, König 1936)**.** Let $t = (V, \rightarrow, \lambda)$ be an infinite labelled tree that is finite branching. Then t has an infinite path.

Proof. Suppose $t = (V, \to, \lambda)$ is an infinite labelled tree that is finite branching. First of all, notice that every node in V belongs to some path. Suppose there are only finitely many paths through t. Since t is infinite, this directly implies that there is an infinite path through t. Next, suppose that there are infinitely many paths through t. We show how to construct an infinite path. Recall that $\to$ is reflexive and transitive. For $u \in V$ let $Up(u) := \{v \in V | u \to v\}$ be the up-set of u. Let $u_0 \in V$ be the lower-most node at which branching occurs and suppose that u_0 has $k > 1$ children which we denote by $v_1^0, ..., v_k^0$. Since u_0 is the first node at which branching occurs, every path in t passes through u_0. That is, there are infinitely many paths passing through u_0. Since u_0 has only finitely many children, there exists a child $v_{i_0}^0$ of u_0 for $1 \leq i_0 \leq k$ such that infinitely many paths in t pass through $v_{i_0}^0$. Let u_1 be the lower-most node of $Up(v_{i_0}^0)$ at which branching occurs. Then since infinitely many paths pass through $v_{i_0}^0$ and u_1 is the first node above $v_{i_0}^0$ where branching occurs, there are infinitely many paths passing through u_1. So by the same argument as before there exists a child $v_{i_1}^1$ of u_1 such that infinitely many paths pass through $v_{i_1}^1$. By iterating this argument we obtain an infinite sequence of natural numbers $(i_n)_{n\in\omega}$ such that for each $n \in \omega$ the following holds:

1. $v_{i_n}^n \in V$
2. $v_{i_n}^n \to v_{i_{n+1}}^{n+1}$

Therefore let $\mathbb{P}$ be the path which satisfies the property that for all $n \in \omega$ there exists $j \in \omega$ such that $\mathbb{P}(j) = v_{i_n}^n$. By construction $\mathbb{P}$ is an infinite path through t. □

Corollary 4.2.2. Let $t = (V, \to, \lambda)$ be an infinite tableau. Then t has an infinite path.

Next, we prove a similar result that states that whenever there are infinitely many traces through a path, then there exists an infinite trace through that path. Notice that this is not a corollary of König's Lemma, as the set of traces through a path is not a tree. Nevertheless, the proof of the lemma is very similar.

Lemma 4.2.3. Let $t = (V, \to, \lambda)$ be a tableau for Γ and let $\mathbb{P}$ be a path through t. If there are infinitely many traces through $\mathbb{P}$, then there is an infinite trace through $\mathbb{P}$.

Proof. Let $t = (V, \to, \lambda)$ be a tableau for Γ and let $\mathbb{P}$ be a path through t such that there are infinitely many traces through $\mathbb{P}$. Since every trace starts in a formula in Γ and Γ is finite, there exists a formula $\varphi \in \Gamma$ from which infinitely many traces through $\mathbb{P}$ start. Notice that for any $n \in \omega$ there are only finitely many possibilities to build different traces through $\mathbb{P}$ starting in φ in n steps. Therefore, there are infinitely many traces starting in φ whose length is greater that n for any $n \in \omega$. This implies that $\mathbb{P}$ is an infinite path. Given two traces that are identical in the first n steps and longer than n, observe that the only case in which these traces might differ from each other in the $n+1$-th step is when the n-th formula is of the form $\psi_0 \wedge \psi_1$ and the rule applied is $(\wedge)$ such that the next node is labelled by ψ_0 and ψ_1. Let T be the set of all traces through $\mathbb{P}$ that start in φ. Suppose $tr = tr(0)tr(1)tr(2)...$ is a trace in T. We call $tr(0)tr(1)...tr(n)$ the n-th initial segment of tr (assuming that the length of tr is greater than n). By assumption $tr(0) = \varphi$ for all traces $tr \in T$. Let i_0 be the least natural number such that the i_0-th initial segment of every trace in T which is longer than i_0 is identical but some traces differ in the $i_0 + 1$-th component. Thus the rule applied at the node $\mathbb{P}(i_0)$ is $(\wedge)$ which decomposes $tr(i_0)$ in two formulas, say in φ_0^0 and φ_0^1. Since there are infinitely many traces starting in φ which are longer than i_0, there are infinitely many traces tr in T which pass through $\varphi_0^{j_0}$ for $j_0 \in \{0, 1\}$. Let T_0 be the set of all these traces. Now let $i_1 > i_0$ be the least natural number such that the i_1-th initial segment of every trace in T_0 which is longer than i_1 is identical but some traces differ in the $i_1 + 1$-th component. By the same argument we find a formula $\varphi_1^{j_1}$ such that there are infinitely many traces tr in T_0 which are identical in the $i_1 + 1$-th initial segment and $tr(i_1 + 1) = \varphi_1^{j_1}$. By iterating this argument we therefore obtain an infinite sequence of tuples

$$\langle i_n, \varphi_n^{j_n} \rangle_{n \in \omega}$$

where $(i_n)_{n \in \omega}$ is an infinite sequence of natural numbers such that $i_n < i_{n+1}$ for all $n \in \omega$ and $(\varphi_n^{j_n})_{n \in \omega}$ is an infinite sequence of subformulas of φ. Let tr be the trace which satisfies the following properties:

1. $tr(0) = \varphi$
2. For all $n \in \omega$ it holds that $tr(i_n + 1) = \varphi_n^{j_n}$.

Observe that tr is a well-defined trace through $\mathbb{P}$ and infinite. □

As an immediate corollary of König's Lemma and its close cousin we obtain the following result:

Proposition 4.2.4. Every tableau $t = (V, \to, \lambda)$ for $\Gamma \subseteq \Sigma_1^\mu$ is finite.

Proof. Suppose $t = (V, \to, \lambda)$ is a tableau for $\Gamma \subseteq \Sigma_1^\mu$ and let $\mathbb{P}$ be a path through t. According to our observation at the beginning of the section every trace through $\mathbb{P}$ is finite, as every trace is starting in a Σ_1^μ-formula. So by lemma 4.2.3 there are only finitely many traces through $\mathbb{P}$. Since each of these traces is finite, the path $\mathbb{P}$ is finite as well. As $\mathbb{P}$ was arbitrary, we conclude that every path through t is finite. By König's Lemma the tableau t is finite. □

Theorem 4.2.5 (Finite model property for Σ_1^μ)**.** If $\Gamma \subseteq \Sigma_1^\mu$ is satisfiable, then it is satisfiable in a finite model.

Proof. Suppose $\Gamma \subseteq \Sigma_1^\mu$ is satisfiable. By Theorem 3.4.1 (stating that **T** is complete) Γ has a tableau $t = (S, \to, \lambda)$ which is finite according to the previous proposition. Hence, by following the construction of the model in the soundness proof for **T**, we obtain a finite tree model in which every formula of Γ is satisfied at the root. Therefore Γ is satisfiable in a finite model. □

4.3 Circular tableaux for $\Sigma_1^\mu \cup \Pi_1^\mu$

From now on, sequents are assumed to be finite subsets of $\Sigma_1^\mu \cup \Pi_1^\mu$. As mentioned in the introduction, circular tableaux are finite labelled trees that unfold into infinitary regular tableaux. These trees are generated by the rules of the system **T**. The main difference to tableaux is that instead of building infinitary branches by unfolding fixed point formulas, we are allowed to end such branches after finitely many steps when we reach a suitable repetition. A repetition is thereby a pair of nodes $\langle u', u\rangle$ in the same branch such that u' and u satisfy certain properties, that ensure that the unfolding of the tree is a tableaux. In this section we introduce the circular tableaux system **CT**. Moreover, we consider some examples and discuss the relevance of the restriction to $\Sigma_1^\mu \cup \Pi_1^\mu$ for the soundness of **CT**.

Table 4.1: The circular tableaux system **CT**

$$\frac{\Gamma, \varphi_0, \varphi_1}{\Gamma, \varphi_0 \wedge \varphi_1}\ (\wedge) \qquad \frac{\Gamma, \varphi_0}{\Gamma, \varphi_0 \vee \varphi_1}\ (\vee)_0 \qquad \frac{\Gamma, \varphi_1}{\Gamma, \varphi_0 \vee \varphi_1}\ (\vee)_1$$

$$\frac{\Gamma, Z}{\Gamma, \mu Z.\varphi(Z)}\ (\mu) \qquad \frac{\Gamma, Z}{\Gamma, \nu Z.\varphi(Z)}\ (\nu) \qquad \frac{\Gamma, \varphi(Z)}{\Gamma, Z}\ (Z)$$

$$\frac{\Gamma, \varphi_1 \quad \dots \quad \Gamma, \varphi_n}{\Box\Gamma, \Diamond\varphi_1, \dots, \Diamond\varphi_n, \Theta}\ (mod) \qquad (\Theta \subseteq Lit \text{ consistent})$$

Definition 4.3.1. The circular tableaux system **CT** consists of the same rules as the tableaux system **T** and is depicted in table 4.1.

Principal and *residual* formulas of a rule are defined as for tableaux, see section 3.3.

Definition 4.3.2. A *circular pre-tableau* for $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ is a finite tree $t = (V, \to, \lambda)$ with root r_t which is generated by the rules in table 4.1 such that:

1. $\lambda(r_t) = \Gamma$
2. every leaf $u \in V$ is labelled either by a sequent of the form $\Box\Delta, \Diamond\Lambda, \Theta$ where $\Theta \subseteq Lit$ and
   (a) $\Lambda = \emptyset$ or
   (b) Θ is inconsistent

   or by a sequent Ω such that there exists a distinguished node $u' \in V$ from which u is reachable and $\lambda(u') = \lambda(u)$. We call u' the *associated node of u.*

We call a leaf which is labelled by a sequent of the form $\Box\Delta, \Diamond\Lambda, \Theta$ a *leaf of type 1* and a leaf which is not of type 1 a *leaf of type 2*.

Next, we define the notion of a path in such a way, that a path which reaches a leaf of type 2 can be continued at its associated node. This allows paths to be infinite, despite the finiteness of a circular pre-tableau.

Definition 4.3.3. Let $t = (V, \rightarrow, \lambda)$ be a circular pre-tableau with root r_t. A *path* $\mathbb{P}$ through t is a (possibly infinite) sequence of nodes $\mathbb{P}(0)\mathbb{P}(1)\mathbb{P}(2)...$ with $\mathbb{P}(0) = r_t$ such that for all $i \in \omega$:

1. If $\mathbb{P}(i)$ is not a leaf, then $\mathbb{P}(i) \rightarrow \mathbb{P}(i+1)$.
2. If $\mathbb{P}(i)$ is a leaf of type 1, then the path ends at $\mathbb{P}(i)$.
3. If $\mathbb{P}(i)$ is a leaf of type 2 and $j < i$ such that $\mathbb{P}(j)$ is the associated node of $\mathbb{P}(i)$, then $\mathbb{P}(j) \rightarrow \mathbb{P}(i+1)$.

Definition 4.3.4. Let $t = (V, \rightarrow, \lambda)$ be a circular pre-tableau for $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ and let $\mathbb{P}$ be a path through t. A finite sequence of formulas $\varphi_0, \varphi_1, ..., \varphi_n$ is a *finite trace* through $\mathbb{P}$ if

1. $\varphi_i \in \lambda(\mathbb{P}(i))$ for each $i \leq n$
2. $\varphi_{i+1} = \varphi_i$ if φ_i is not principal in the rule from $\mathbb{P}(i)$ to $\mathbb{P}(i+1)$, otherwise φ_{i+1} is (one of) the residual(s) of the rule.

An infinite sequence of formulas $\varphi_0, \varphi_1, ...$ is an *infinite trace* if every initial segment of the sequence is a finite trace.

Notice that a formula labelling a leaf of type 2 can belong to several different traces, as there might exist a node between the associated node of the leaf and the leaf at which the trace splits. However, each trace to which the formula belongs to starts in the same formula at the root.

Definition 4.3.5. A *circular tableau* for $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ is a circular pre-tableau $t = (V, \rightarrow, \lambda)$ for Γ where the following holds:

1. Every leaf of type 1 is labelled by $\Box\Delta, \Theta$ where $\Theta \subseteq Lit$ is consistent.
2. Every leaf u of type 2 has the following two properties:
   (a) The leaf u is labelled by Ω where each $\psi \in \Omega$ belongs to a trace starting in a Π_1^μ-formula labelling the root.
   (b) There is an application of the rule (mod) between u' and u.

Leafs of type 1 that are labelled by $\Box\Delta, \Theta$ for $\Theta \subseteq Lit$ consistent are called *axiomatic leafs* and leafs of type 2 which fulfil the requirements above are called *non-axiomatic leafs*. Notice that if an infinite trace starts in a Π_1^μ-formula, it is automatically a ν-trace. A circular tableaux is therefore a finite tree, where each leaf is either axiomatic or can be identified with an earlier node in such a way, that every formula labelling the non-axiomatic leaf belongs to a ν-trace and the path passes through a (mod)-rule between the associated node and the leaf. The requirement to pass through a (mod)-rule ensures that there is some progress between the two repetitions (instead of just having a 'silly' repetition). The requirement that every formula that labels the non-axiomatic leaf belongs to a ν-trace ensures that when unfolding the circular tableaux into an infinitary pre-tableau, every infinite trace is a ν-trace.

Example 4.3.6. The following is a circular tableau for the formula $\nu Y.\Diamond Y \wedge \Diamond(P \vee Q)$:

$$\dfrac{\dfrac{\dfrac{\dfrac{Y \qquad \dfrac{P}{P \vee Q}\,(\vee)_0}{\Diamond Y, \Diamond(P \vee Q)}\,(mod)}{\Diamond Y \wedge \Diamond(P \vee Q)}\,(\wedge)}{Y}\,(Y)}{\nu Y.\Diamond Y \wedge \Diamond(P \vee Q)}\,(\nu)$$

The left leaf labelled by Y is non-axiomatic with its associated node being the second node from the bottom upwards. Notice that there is an application of (mod) between the associated node and the leaf. The right leaf is axiomatic, as it is labelled by consistent literals only. By unfolding the circular tableau over the non-axiomatic leaf we obtain the following infinite pre-tableau:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\text{etc.}}{Y} \qquad \dfrac{P}{P \vee Q}\,(\vee)_0}{\Diamond Y, \Diamond(P \vee Q)}\,(mod)}{\Diamond Y \wedge \Diamond(P \vee Q)}\,(\wedge)}{Y}\,(Y) \qquad \dfrac{P}{P \vee Q}\,(\vee)_0}{\Diamond Y, \Diamond(P \vee Q)}\,(mod)}{\Diamond Y \wedge \Diamond(P \vee Q)}\,(\wedge)}{Y}\,(Y)}{\nu Y.\Diamond Y \wedge \Diamond(P \vee Q)}\,(\nu)$$

Observe that this is a tableaux. There are infinitely many leafs which are all labelled by consistent literals. Furthermore, there is a single infinite path that contains a single infinite trace which passes through the ν-variable Y infinitely often. As there are no other variables present, the infinite trace is a ν-trace. The example illustrates the idea of unfolding circular tableaux into infinitary tableaux. Notice also that the underlying tree of this infinitary tableau is regular.

Observe that it is not required to choose the first suitable repetition in a branch to be identified as non-axiomatic leaf and associated node. For instance, if we take the infinite tableau in the example above and prune it at the third occurrence of Y (directly below etc.), we obtain a different circular tableaux. This implies that a sequent can have infinitely many different circular pre-tableaux.

We finish this section with a brief discussion why the the restriction to $\Sigma_1^\mu \cup \Pi_1^\mu$ is relevant. The main reason concerns the unfolding of circular tableaux. We require formulas which label non-axiomatic leafs to belong to traces starting in Π_1^μ-formulas. This implies that in the unravelled tree, every infinite trace is a ν-trace. When we have fixed point alternation however, such an easy characterization of non-axiomatic leafs is not possible. The main problem is best illustrated by the following example. Consider the formula $\varphi = \nu Y.\mu Z.\Diamond Y \vee \Diamond Z$. Below is a tableaux for φ:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\text{etc.}}{Y}}{\Diamond Y}\,(mod)}{\Diamond Y \vee \Diamond Z}\,(\vee)_0}{Z}\,(Z)}{\Diamond Z}\,(mod)}{\Diamond Y \vee \Diamond Z}\,(\vee)_1}{Z}\,(Z)}{\mu Z.\Diamond Y \vee \Diamond Z}\,(\mu)}{Y}\,(Y)}{\nu Y.\mu Z.\Diamond Y \vee \Diamond Z}\,(\nu)$$

In this tableau we keep alternating between unfolding and regenerating the least fixed point variable Z and the greatest fixed point variable Y. That is, the only infinite trace passes through both Y and Z infinitely often. Since Y subsumes Z, the trace is a ν-trace and the pre-tableau a tableau. If we want to turn this tableau into a circular tableau, we have to choose at which repetition we prune the branch. Previously, it did not matter which repetition we choose as long as the distinguished node is only labelled by formulas belonging to traces that start in Π_1^μ-formulas. Here, it does. If we decide to prune the tableau at the node labelled by the second occurrence of Z, we create a circular pre-tableau which when unfolded turns into an infinitary pre-tableau that contains an infinite μ-trace. If we choose the second occurrence of Y, then the circular pre-tableau which is generated is unfolded into the infinitary tableau above. Thus, for formulas containing proper fixed point alternation, we require more refined conditions when a repetition can be used as a non-axiomatic leaf. Afshari and Leigh solved this problem by turning towards proof systems using annotated sequents [1]. As we only consider the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$, our definition suffices.

4.4 Finite model property for $\Sigma_1^\mu \cup \Pi_1^\mu$

In this section we establish the finite model property for the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$. We first prove that the system **CT** is sound and complete with respect to $\Sigma_1^\mu \cup \Pi_1^\mu$. For soundness we proceed as in the soundness proof for **T**. From a circular tableau for Γ, we show how to define a model and then prove, using model checking games, that Γ is satisfiable in some state. As the circular tableau is finite, we are able to define a finite model. Notice that we could establish soundness as well by showing that every circular tableaux unravels into an infinitary tableaux. This would be much easier, but we could not extract the finite model property out of the proof. For completeness, we show how to prune an infinitary tableau into a circular one. Completeness together with soundness then gives us the finite model property, as every satisfiable sequent has a circular tableau by completeness, which then implies that it has a finite model by soundness.

Theorem 4.4.1 (Soundness of **CT** with respect to $\Sigma_1^\mu \cup \Pi_1^\mu$)**.** Suppose $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$. If Γ has a circular tableau, then Γ is satisfiable.

For the proof of the soundness theorem we require the following lemma.

Lemma 4.4.2. Suppose $t = (V, \rightarrow, \lambda)$ is a circular tableau for $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ and v is a non-axiomatic leaf with associated node u. Then each formula $\psi \in \lambda(u)$ which is not a literal or a variable is decomposed in the steps between node u and v. Additionally, each literal in $\lambda(u)$ is eliminated and each variable in $\lambda(u)$ is regenerated at some step between u and v.

Proof. (of the lemma) Suppose $t = (V, \rightarrow, \lambda)$ is a circular tableau for $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ and v is a non-axiomatic leaf with corresponding node u. By definition there is an application of (*mod*) between the nodes u and v. Let $\psi \in \lambda(u)$ be a formula which is not a literal or a variable. If $\psi = \Box\psi_0$ or $\psi = \Diamond\psi_0$, then ψ is decomposed when the rule (*mod*) is applied. In case $\psi = \psi_0 \vee \psi_1$ or $\psi = \psi_0 \wedge \psi_1$ or $\psi = \sigma Y.\psi_0$ for $\sigma \in \{\mu, \nu\}$, then there must be an application of $(\vee), (\wedge)$ or (σ) between u and v as otherwise the rule (*mod*) cannot be applied. Therefore each formula is decomposed between u and v. With the same argument, if ψ is a literal, then ψ is eliminated when the rule (*mod*) is applied and if ψ is a variable Y, there must be an application of the rule (Y) before the application of (*mod*) thus regenerating Y. $\Box$

Proof. (of the theorem) Let $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ and let $t = (V, \rightarrow_t, \lambda)$ be a circular tableau for Γ with root r_t. We define a transition system $T = (S, \rightarrow_T, \rho)$ and a map $\tau : V \longrightarrow S$, such that the following holds:

- ▷ $\tau(r_t) = s_0$ for $s_0 \in S$.
- ▷ Suppose $v \rightarrow_t u$. If the rule applied at node v was (mod), then $\tau(v) \neq \tau(u)$ and $\tau(v) \rightarrow_T \tau(u)$, Otherwise, $\tau(v) = \tau(u)$.
- ▷ If v is a non-axiomatic leaf and u is its corresponding node, then for all $s \in S$, if $\tau(u) \rightarrow_T s$, then $\tau(v) \rightarrow_T s$.

- ▷ $s \in \rho(P)$ if and only if there exists $v \in V$, such that $\tau(v) = s$ and one of the following holds:
  - $P \in \lambda(v)$ or
  - v is a non-axiomatic leaf with associated node u, such that $\tau(u) \in \rho(P)$

The second requirement for the function ρ is necessary due to the fact that a path can loop back from a non-axiomatic leaf to its associated node. When we define corresponding traces and plays, it might therefore happen that the constructed trace has passed through a non-axiomatic leaf and then ends before reaching the first application of (mod). In that case, the current state of the play does not correspond to the current node in the tree. With the additional requirement we ensure that finite plays of this form are still won by Verifier.

Observe that the function ρ is well-defined. That is, there exists no node $s \in S$ such that both P and $\neg P$ hold in s. Therefore, T is a well-defined transition system. Notice that since the circular tableau t is finite, the transition system T is finite as well. Let φ be an arbitrary formula in Γ and consider the model checking game $\mathcal{G}_{\emptyset}^{T}(s_0, \varphi)$. We show that Verifier has a memoryless winning strategy, which is defined by using information from t. In particular, we show that every play in the game corresponds to a trace through the circular tableau. Recall that every play of $\mathcal{G}_{\emptyset}^{T}(s_0, \varphi)$ starts in position (s_0, φ_0) where $\varphi_0 = \varphi$. We restrict our attention to traces that start in $\varphi_0 = \varphi$. Then the initial segment (s_0, φ_0) of every play corresponds to the initial segment φ_0 of every trace considered. Suppose we have an initial segment of a play

$$(s_0, \varphi_0), (s_1, \varphi_1), ..., (s_n, \varphi_n)$$

which corresponds to the initial segment of a trace

$$\varphi_0, ..., \varphi_0, \varphi_1, ..., \varphi_1, ..., \varphi_n$$

such that $\varphi_n \notin Lit$. We show how to extend the play and the trace:

Case 1: It is Verifier's move. This implies that φ_n is either $\psi_0 \vee \psi_1$ or $\Diamond\psi$.

- ▷ Suppose $\varphi_n = \psi_0 \vee \psi_1$ labels the node v. There are two cases: In the first case $\tau(v) = s_n$ while in the second $\tau(v) \neq s_n$. Let us consider the first case: Recall that by the previous lemma every formula that labels a non-axiomatic leaf and its associated node is decomposed in the steps in between. Therefore, since t is a circular tableau, the formula $\psi_0 \vee \psi_1$ is either decomposed at some node after v or at some node before v (namely, in case v belongs to a branch leading to a non-axiomatic leaf, such that there is no application of $(\vee)_i$ to $\psi_0 \vee \psi_1$ after v). The first case is identical to the soundness proof for tableaux. Therefore we only consider the second case. Suppose that there exists no node after v at which $\psi_0 \vee \psi_1$ is decomposed. This implies that the path to which v belongs leads to a non-axiomatic leaf w with associated node w' occurring earlier than v

such that $\psi_0 \vee \psi_1$ is decomposed at some node u between w' and v where u's child u' is labelled by ψ_i for $i \in \{0,1\}$. By definition of paths and traces through circular tableaux we can therefore extend the trace to

$$\varphi_0, ..., \varphi_0, \varphi_1, ..., \varphi_1, \cdots \underbrace{\varphi_n, ..., \varphi_n}_{k-times}, \varphi_{n+1}$$

where $k = l + j + 1$ where l is the number of steps between v and w, j is the number of steps between w' and u and $\varphi_{n+1} = \psi_i$. Meanwhile, the play can be extended to

$$(s_0, \varphi_0), (s_1, \varphi_1), ..., (s_n, \varphi_n), (s_{n+1}, \varphi_{n+1})$$

where $(s_{n+1}, \varphi_{n+1}) = (s_n, \psi_i)$. Notice that in this case the node u' which is labelled by ψ_i is not mapped to s_n. Therefore, this is exactly the second case we have to consider: If $\tau(v) \neq s_n$, then v occurs between the associated node w' and the first application of (mod) after w', where the trace has already passed through the non-axiomatic leaf w at least once. But in that case we can still extend the trace and the play in the same way as before. The only reason why we have to keep track of nodes and their corresponding states with respect to τ is when the current formula of the play is a diamond- or box-formula: In that case one of the players has to choose the next position which requires that the current state s_n sees a successor. The existence of a successor is guaranteed by the correspondence between the node in the tableau and the current state: If $\tau(v) = s_n$, the current formula is $\Diamond\psi$ and some steps later there is the node u at which the rule (mod) is applied, then there exists a child node u' labelled by ψ and the definition of the transition system ensures that $s_n \rightarrow_T \tau(u')$ and so that Verifier can extend the play. In the situation that $\tau(v) \neq s_n$, we know that $s_n = \tau(w)$ where w is the non-axiomatic leaf and v occurs between the associated node w' and the first application of (mod) above w' at node u. So when (mod) is applied one of the players has to extend the play. But by construction of the transition system, $\tau(w) = s_n$ has a transition to every state $\tau(w') = \tau(u)$ has a transition to and so there exists a state which can be used to extend the play. In the following case distinctions we omit this special case, as the same argument as just given suffices.

- ▷ Suppose $\varphi_n = \Diamond\psi$ where φ_n labels node v which is mapped to s_n by τ. We only consider the case where there is no node after v at which the rule (mod) is applied. Since t is a circular tableau, v therefore belongs to a path leading to a non-axiomatic leaf w with associated node w' occurring below v. Thus $\Diamond\psi$ labels every node between v and w as well as w and w' and is decomposed at some node u between w' and v. This implies that the rule applied at u is (mod) and there exists a child u' of u labelled by ψ. By definition of paths and traces we can therefore extend the trace to

$$\varphi_0, ..., \varphi_0, \varphi_1, ..., \varphi_1, ..., \underbrace{\varphi_n, ..., \varphi_n}_{k-times}, \varphi_{n+1}$$

where $k = l + j + 1$ where l is the number of steps between v and w, j is the number of steps between w' and u and $\varphi_{n+1} = \psi_i$. Notice that since there is no application of (mod) between v and w, we have that $\tau(v) = \tau(w) = s_n$. Moreover, there is also no application of (mod) between w' and u (as otherwise $\Diamond\psi$ would be decomposed before u) which implies that $\tau(w') = \tau(u)$. By construction we have that $\tau(u) \rightarrow_T \tau(u')$ and so also $\tau(w') \rightarrow_T \tau(u')$. But then by construction of the transition system T we also have that $\tau(w) \rightarrow_T \tau(u')$ and so that $s_n \rightarrow_T \tau(u')$. We therefore extend the play to

$$(s_0, \varphi_0), (s_1, \varphi_1), ..., (s_n, \varphi_n), (s_{n+1}, \varphi_{n+1})$$

where $(s_{n+1}, \varphi_{n+1}) = (\tau(u'), \psi)$.

Case 2: It is Refuter's move. Then φ_n is either $\psi_0 \wedge \psi_1$ or $\Box\psi$. We show that no matter what choices Refuter takes, the trace and play can be extended accordingly. As in case 1, in both cases we have two distinguish whether the current node v is mapped to the current state s_n and if so, whether the respective formula of the case is decomposed before reaching a leaf or not. Once again we only consider the case where $\tau(v) = s_n$ and the respective formula is not decomposed above v.

▷ Suppose $\varphi_n = \psi_0 \wedge \psi_1$ and suppose Refuter chooses to extend the play to

$$(s_0, \varphi_0), (s_1, \varphi_1), ..., (s_n, \varphi_n), (s_{n+1}, \varphi_{n+1})$$

where $(s_{n+1}, \varphi_{n+1}) = (s_n, \psi_i)$ for $i \in \{0, 1\}$. Suppose that φ_n labels the node v which is mapped to s_n by τ such that $\psi_0 \wedge \psi_1$ is not decomposed in any node above v. This implies that v belongs to a path leading to a non-axiomatic leaf w with associated node w', both labelled by $\psi_0 \wedge \psi_1$, and $\psi_0 \wedge \psi_1$ is decomposed at some node u between w' and v. Notice that the child u' of u is labelled by ψ_0 and ψ_1. Thus by definition of paths and traces through circular tableaux we can extend the trace to

$$\varphi_0, ..., \varphi_0, \varphi_1, ..., \varphi_1, ..., \underbrace{\varphi_n, ..., \varphi_n}_{k-times}, \varphi_{n+1}$$

where $k = l + j + 1$ where l is the number of steps between v and w, j is the number of steps between w' and u and $\varphi_{n+1} = \psi_i$.

▷ Suppose $\varphi_n = \Box\psi$ and φ_n labels the node v which is mapped to s_n by τ. Moreover suppose that $\Box\psi$ is not decomposed above v. We assume Refuter can extend the play (otherwise Verifier trivially wins). So there exists a state s in the transition system T such that $s_n \rightarrow_T s$ and Refuter chooses to extend the play to

$$(s_0, \varphi_0), (s_1, \varphi_1), ..., (s_n, \varphi_n), (s_{n+1}, \varphi_{n+1})$$

where $(s_{n+1}, \varphi_{n+1}) = (s, \psi)$. The construction of the transition system T implies that v belongs to a path leading to a non-axiomatic leaf w with associated node w' and $\Box\psi$ is decomposed at some node u between w' and v (otherwise Refuter could not extend the play). The rule applied at u is therefore (mod) and every child of u is labelled by ψ. In particular there exists a child u' of u such that $\tau(u') = s$. As u' is labelled by ψ, we extend the trace to

$$\varphi_0, ..., \varphi_0, \varphi_1, ..., \varphi_1, ..., \underbrace{\varphi_n, ..., \varphi_n}_{k-times}, \varphi_{n+1}$$

where $k = l + j + 1$ where l is the number of steps between v and w, j is the number of steps between w' and u and $\varphi_{n+1} = \psi$.

Case 3: It is a neutral move. This implies that φ_n is either $\mu Z.\psi(Z)$, $\nu Z.\psi(Z)$ or Z. In all three cases it follows directly that both the play and the trace can be extended accordingly.

In case 1 we provided a memoryless strategy for Verifier by following some trace through t. In case 2 we showed that no matter what choices Refuter takes to extend the play, the trace can always be extended in such a way, that the play and the trace are corresponding. Therefore, if Verifier plays according to the strategy specified in case 1, every play corresponds to some trace through t. We show that case 1 defines a winning strategy for Verifier. Suppose we have a finite play $(s_0, \varphi_0), ..., (s_n, \varphi_n)$ where Verifier uses the above described strategy. Let $\varphi_0, ..., \varphi_0, ..., \varphi_n$ be the corresponding trace and v the node of the circular tableau labelled by φ_n at which the trace ends. By definition φ_n is either a literal, a boxed formula where Refuter could not extend the play or a diamond formula where Verifier could not extend the play.

1. Suppose $\varphi_n \in Lit$. We distinguish two cases. First, suppose that $\tau(v) = s_n$. If $\varphi_n = P$, then $P \in \lambda(v)$ and so $s_n \in \rho(P)$. If $\varphi_n = \neg P$, then by lemma 3.4.2 it follows that $P \notin \lambda(v)$, which in turn implies that $s_n \notin \rho(P)$ (by a similar argument as in the soundness proof for **T**). Second, suppose that $\tau(v) \neq s_n$. Then there exists a node u such that $\tau(u) = s_n$ which is a non-axiomatic leaf with associated node u' and $\tau(u') = \tau(v)$. If $\varphi_n = P$, then $P \in \lambda(v)$ and since $\tau(u') = \tau(v)$ it follows that $\tau(u') \in \rho(P)$. Therefore by definition of the function ρ we have that $s_n \in \rho(P)$. The case for $\varphi_n = \neg P$ is similar. We conclude that in all cases Verifier wins.

2. Suppose $\varphi_n = \Box\psi$ and Refuter could not extend the play. This directly implies that Verifier wins.

3. Suppose $\varphi_n = \Diamond\psi$ and Verifier could not extend the play. If this was the case, the corresponding trace ends in the formula $\Diamond\psi$. Now suppose it is possible to extend the trace to ψ. In that case Verifier could have extended the play according to the strategy, as extending the trace implies that there exists a successor node of s_n in the transition system labelled by ψ. Hence the trace cannot be extended, which implies that there is some leaf of type 1 labelled by $\Diamond\psi$, contradicting our assumption that t is a circular tableau. Therefore this case cannot occur.

Thus whenever we have a finite play in $\mathcal{G}^T_\emptyset(s_0, \varphi)$, Verifier wins. Now suppose we have an infinite play $(s_0, \varphi_0), (s_1, \varphi_1), ...$ corresponding to the infinite trace $tr = \varphi_0, ..., \varphi_0, \varphi_1, ..., \varphi_1, ...$ trough an infinite path $\mathbb{P}$ in t. Since t is a finite tree, a path can only be infinite if it passes through some non-axiomatic leafs infinitely often and every time loops back to its associated node. Recall that we defined non-axiomatic leafs to be labelled by formulas belonging to traces starting in Π^μ_1-formulas only. That is, every infinite trace has to start in a Π^μ_1-formula. Therefore every infinite trace through t is a ν-trace and so in particular tr is a ν-trace. Hence the unique variable that occurs infinitely often in tr and which subsumes every other infinitely often occurring variable is a ν-variable. Since the trace corresponds to the play, we have that unique variable occurring infinitely often in the play and subsuming every other infinitely often occurring variable is a ν-variable as well and so the play is won by Verifier. Hence Verifier wins every infinite play as well. Together we conclude that the strategy provided in case 1 is a memoryless winning strategy for Verifier. The Fundamental Semantic Theorem thus implies that $s_0 \in [\![\varphi]\!]^T_\emptyset$. As we chose φ to be an arbitrary formula in Γ, we have that $s_0 \in [\![\bigwedge \Gamma]\!]^T_\emptyset$ and so that Γ is satisfiable. We conclude that the circular system **CT** is sound with respect to $\Sigma^\mu_1 \cup \Pi^\mu_1$. □

The next goal is to prove completeness of **CT**. Our proof strategy is to show how to prune infinitary tableaux to obtain circular tableaux. For this, we first have to establish that every infinite branch of a tableau contains a suitable repetition. We use the following notation. Let $t = (V, \to, \lambda)$ be a tableau and $\mathbb{P} = \mathbb{P}(0)\mathbb{P}(1)\mathbb{P}(2)...$ a path through t. We say that the node $\mathbb{P}(n)$ is labelled by a *Π^μ_1-formula* if there exists a formula $\psi \in \lambda(\mathbb{P}(n))$, such that ψ belongs to a trace starting in $\tilde{\psi}$ and $\tilde{\psi} \in \Pi^\mu_1$.

Lemma 4.4.3. Let $\Gamma \subseteq \Sigma^\mu_1 \cup \Pi^\mu_1$ and suppose that $t = (V, \to, \lambda)$ is a tableau for Γ. Let $\mathbb{P} = \mathbb{P}(0)\mathbb{P}(1)\mathbb{P}(2)...$ be an infinite path through t. There exists $n \in \omega$ such that for all $m > n$ every formula labelling the node $\mathbb{P}(m)$ is a Π^μ_1-formula.

Proof. First of all, if $\Gamma \subseteq \Pi^\mu_1$, then the lemma is trivially true. Similarly, if $\Gamma \subseteq \Sigma^\mu_1$, then by proposition 4.2.4, there are no infinite paths through the tableau t. So suppose $\Gamma = \{\varphi_1, ..., \varphi_k, \psi_1, ..., \psi_l\}$ where $\varphi_1, ..., \varphi_k \in \Sigma^\mu_1$ and $\psi_1, ..., \psi_l \in \Pi^\mu_1 - \Sigma^\mu_0$ and suppose that $t = (V, \to, \lambda)$ is a tableau for Γ. Let $\mathbb{P} = \mathbb{P}(0)\mathbb{P}(1)\mathbb{P}(2)...$ be an infinite path through t. Consider $\varphi_i \in \Gamma$. Suppose there are infinitely many traces through $\mathbb{P}$ starting in φ_i. Then by lemma 4.2.3 there is an infinite trace starting in φ_i. As this trace is a μ-trace we conclude that t is not a tableau, which is a contradiction. Therefore there are only finitely many traces starting in φ_i for all $1 \leq i \leq k$. Since each of these traces is finite, there exists a longest trace, which has length say n. Finally notice that every formula labelling some node of $\mathbb{P}$ belongs to a trace. So since the longest trace starting from a Σ^μ_1-formula in Γ has length n, all nodes $\mathbb{P}(m)$ for $m > n$ are labelled by Π^μ_1-formulas only. □

Lemma 4.4.4. Let $t = (V, \to, \lambda)$ be a tableau for $\Gamma \subseteq \Sigma^\mu_1 \cup \Pi^\mu_1$ and $\mathbb{P} = \mathbb{P}(0)\mathbb{P}(1)...$ an infinite path through t. Let $n \in \omega$ be an arbitrary natural number. There exist a pair $\langle i, j\rangle$ of natural numbers such that $n < i < j$, $\lambda(\mathbb{P}(i)) = \lambda(\mathbb{P}(j))$ and there is an application of (mod) between $\mathbb{P}(i)$ and $\mathbb{P}(j)$.

Proof. Let $\mathbb{P} = \mathbb{P}(0)\mathbb{P}(1)\mathbb{P}(2)...$ be an infinite path through $t = (V, \rightarrow, \lambda)$ where t is a tableau for $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ and fix a natural number n. By proposition 3.3.7 every infinite path passes through a (mod)-rule infinitely often. So there are infinitely many nodes $\mathbb{P}(i_0), \mathbb{P}(i_1), ...$ where the rule applied at $\mathbb{P}(i_l)$ is (mod). In particular, there are infinitely many such nodes with $i_l > n$. Next, since the root of t is labelled by Γ, each node $\mathbb{P}(m)$ in the path is labelled by some finite set $\Delta \subseteq Sub(\Gamma)$. Suppose $|Sub(\Gamma)| = k$. Then there are 2^k different subsets of $Sub(\Gamma)$. Hence there must be some $\Delta \subseteq Sub(\Gamma)$ that labels infinitely many nodes in the path $\mathbb{P}$. In particular, Δ labels infinitely many nodes $\mathbb{P}(i)$ with $i > n$. Suppose $\mathbb{P}(i)$ is the first node that is labelled by Δ such that $i > n$. Then since there are infinitely many nodes $\mathbb{P}(i_l)$ where the (mod)-rule is applied, there exists $l \in \omega$ such that $i < i_l$. Since there are also infinitely many nodes labelled by Δ, there exists $j \in \omega$ such that $i_l < j$ and $\lambda(\mathbb{P}(j)) = \Delta$. Therefore there exists a pair $\langle i, j \rangle$ of natural numbers such that $n < i < j$, $\lambda(\mathbb{P}(i)) = \lambda(\mathbb{P}(j))$ and there is an application of (mod) between $\mathbb{P}(i)$ and $\mathbb{P}(j)$. □

Theorem 4.4.5 (Completeness of **CT** with respect to $\Sigma_1^\mu \cup \Pi_1^\mu$)**.** Let $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$. If Γ is satisfiable, then Γ has a circular tableau.

Remark 4.4.6. Recall that we defined infinite paths and traces through a circular tableau, by allowing a path which goes through a leaf of type 2 to be continued at its associated node. In the following proof we understand paths and traces in the original sense, namely a path ends in a leaf and there are no loops back to corresponding nodes.

Proof. Let $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ be satisfiable. By theorem 3.4.1 (stating that the tableaux system **T** is complete) Γ has a tableau $t = (V, \rightarrow, \lambda)$. We show how to turn t into a circular tableau $t' = (V', \rightarrow', \lambda')$. First of all, if t is a finite tableau, then t is also a circular tableau. Thus suppose that t is infinite. This implies (by König's Lemma) that t has an infinite path. By lemma 4.4.3 for each infinite path $\mathbb{P}$ there exists a natural number n such that every node after $\mathbb{P}(n)$ is labelled by Π_1^μ-formulas only. Lemma 4.4.4 implies that there exists a suitable repetition in $\mathbb{P}$ above $\mathbb{P}(n)$. That is, there exist $i, j \in \omega$ such that $n < i < j$, $\lambda(\mathbb{P}(i)) = \lambda(\mathbb{P}(j))$ and there is an application of (mod) between $\mathbb{P}(i)$ and $\mathbb{P}(j)$. As natural numbers are well-founded, there exists a first such repetition. So given t, we prune every infinite path at its first repetition of this form. This gives us a tree t' which has only finite paths. We claim that t' is a circular tableau for Γ:

1. Since t is a tableau, t' is generated by the rules from table 4.1.

2. By construction every path in t' is finite. Since t' is finite branching, König's Lemma implies that t' is a finite tree.

3. The root of t' is labelled by Γ.

4. Suppose u is a leaf of t'. There are two possibilities:
   (a) The leaf u is also a leaf of t. Then since t is a tableau, u is labelled by $\Box\Delta, \Theta$ where $\Theta \subseteq Lit$ is consistent.
   (b) The leaf u was generated by pruning an infinite path. Then by construction there exists an associated node u' earlier in the path leading to u which is labelled by the same sequent as u and there is an application of the rule (mod) in between. Moreover, u is only labelled by Π_1^μ-formulas.

Therefore, t' is a circular tableau. We conclude that every satisfiable sequent $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ has a circular tableau, which implies that **CT** is complete. $\Box$

As a corollary of the soundness and completeness theorem, we obtain the desired finite model property for $\Sigma_1^\mu \cup \Pi_1^\mu$.

Theorem 4.4.7 (Finite model property for $\Sigma_1^\mu \cup \Pi_1^\mu$)**.** If $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ is satisfiable, then it satisfiable in a finite model.

Proof. Suppose $\Gamma \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ is satisfiable. By the completeness theorem 4.4.5 Γ has a circular tableau $t = (V, \rightarrow, \lambda)$. Following the proof of the soundness theorem 4.4.1 we construct a finite transition system $T = (S, \rightarrow, \rho)$ and a state $s_0 \in S$ such that Γ holds at s_0. Therefore Γ is satisfiable in a finite model. $\Box$

Chapter 5

Infinitary proof systems for the modal mu-calculus

5.1 Introduction

This chapter is devoted to introduce and discuss three different but closely related infinitary proof systems for the modal mu-calculus. All three systems are Gentzen style sequent calculi, where proofs are (possibly) infinite labelled trees that are finite branching. The first system is a sequent calculus called **DT** which is a dualized version of the system **T**. The name **DT** stands for *Dualized Tableaux*. The second proof system is a slight variation of **DT** and is called **DT′**. It consists of the same axioms and rules as **DT** with the exception of the modality rule, where the rather unusual modality rule of **DT** is replaced by a more standard rule which is common to use in proof systems for the modal mu-calculus; see for example [18]. The third proof system turns **DT′** into a two-sided sequent calculus, which is a proof system that works on two-sided sequents instead of the one-sided sequents considered so far. A two-sided sequent is an ordered pair of finite sets of formulas, written $\Gamma \Rightarrow \Delta$. The change towards two-sided sequents is motivated by the later goal to establish Craig interpolation, for which two-sided sequents are more natural to work with. Apart from introducing these three proof systems, this chapter also consists of soundness and completeness proofs for each of them. For the first system **DT**, soundness and completeness is established by using the close connection to the tableaux system **T** and the already established result of **T**'s soundness and completeness. Notice that **T** as a tableaux system is sound and complete with respect to satisfiability, meaning that a sequent is satisfiable if and only if it has a tableau. Proof systems are sound and complete with respect to validity, meaning that a sequent is valid if and only if it has a proof. The soundness and completeness of the remaining two systems is then derived from the soundness and completeness results established for **DT**.

The next two sections 5.2 and 5.3 are devoted to introduce the sequent calculus **DT** and establish its soundness and completeness. The sections 5.4 and 5.5 realize the same for $\mathbf{DT}'$ and the sections 5.6 and 5.7 for **2DT**.

5.2 The sequent calculus DT

We introduce the sequent calculus **DT**. This is a Gentzen style proof system for the modal mu-calculus where proofs are finite branching infinite trees. The system operates on sequents. As before, a sequent is a finite set of L_μ-formulas. Recall that the interpretation of a sequent Γ in the context of the tableaux system **T** is the conjunction over all formulas in Γ. In the presence of proof systems, the interpretation of sequent is the disjunction over its formulas.

Definition 5.2.1. The *interpretation* of a sequent Γ is given by

$$\mathcal{I}(\Gamma) := \bigvee \Gamma$$

A sequent Γ is called *valid* if and only if $\mathcal{I}(\Gamma)$ is valid and *invalid* otherwise.

Table 5.1: The sequent calculus **DT**

$$\frac{}{\Gamma, P, \neg P}\ (\mathrm{A}) \qquad \frac{\Gamma, \varphi_0 \quad \Gamma, \varphi_1}{\Gamma, \varphi_0 \wedge \varphi_1}\ (\wedge) \qquad \frac{\Gamma, \varphi_0, \varphi_1}{\Gamma, \varphi_0 \vee \varphi_1}\ (\vee)$$

$$\frac{\Gamma, Z}{\Gamma, \mu Z.\varphi(Z)}\ (\mu) \qquad \frac{\Gamma, Z}{\Gamma, \nu Z.\varphi(Z)}\ (\nu) \qquad \frac{\Gamma, \varphi(Z)}{\Gamma, Z}\ (Z)$$

$$\frac{\Gamma, \varphi_i}{\Diamond\Gamma, \Box\varphi_1, ..., \Box\varphi_n, \Theta}\ (\Box) \qquad \text{for } \Theta \subseteq \mathit{Lit} \text{ consistent}$$

Definition 5.2.2. The proof system **DT** consists of

1. the axiom (A) and the rules
2. $(\wedge)$,$(\vee)$,$(\Box)$,(μ),(ν) and (Z)

and is depicted in table 5.1.

Similar to the modality rule in **T** the $(\Box)$-rule can only be applied when there is at least one boxed formula in the sequent and $\Theta \subseteq \mathit{Lit}$ is consistent. The set Γ however is allowed to be empty. For the (Z)-rule it is once again assumed that Z identifies $\varphi(Z)$ and that this identification is unique. The notions of *principal* and *residual* formulas are defined as before.

Definition 5.2.3. A *pre-proof* for Γ is a (possibly infinite) tree $t = (V, \rightarrow, \lambda)$ whose root is labelled by Γ and which is built according to the rules depicted in table 5.1, such that every leaf is labelled by an axiom or by a sequent of the form $\Diamond\Delta, \Theta$ for $\Theta \subseteq Lit$.

The condition on leafs ensures that pre-proofs are maximal. A path can only end when it reaches an axiom or a node labelled by a sequent, for which no rule can be applied any more. The definition of a *path* through a pre-proof and of a *trace* through a path are identical to the definitions of path and trace for pre-tableaux. We refer to definition 3.3.6 and definition 3.3.8. Observe that branching in a pre-proof only occurs when the rule $(\wedge)$ is applied. This is different to the tableaux system **T** where branching is caused by the (mod)-rule.

Definition 5.2.4. A *proof* for Γ is a pre-proof $t = (V, \rightarrow, \lambda)$ for Γ such that every leaf is labelled by an axiom and every infinite path contains an infinite ν-trace.

We write $\mathbf{DT} \vdash \Gamma$ if and only if there exists a proof for Γ. Leafs which are labelled by axioms are called *axiomatic leafs*.

5.3 Soundness and completeness of DT

This section establishes the soundness and completeness of **DT**. In doing so we also establish the connection between **DT** and **T** which justifies thinking of **DT** as a dualized version of **T**. The section consists of three subsections. In the first we establish some preliminaries needed throughout the rest of the section. The last two subsections are then devoted to establish soundness and completeness of **DT**.

5.3.1 Preliminaries

In order to connect the system **DT** with the tableaux system **T**, we require to reason about negated formulas. As negation only occurs on propositional level, we introduce a translation D which maps a formula φ onto the formula $D(\varphi)$, such that $\neg\varphi \equiv D(\varphi)$.

Definition 5.3.1. Let φ be an L_μ-formula. The translation D is defined inductively as follows:

$$
\begin{array}{lll}
D(P) & := & \neg P \text{ for all } P \in Prop \\
D(\neg P) & := & P \text{ for all } P \in Prop \\
D(Z) & := & \neg Z \text{ for all } Z \in Var \\
D(\neg Z) & := & Z \text{ for all } Z \in Var \\
D(\varphi \wedge \psi) & := & D(\varphi) \vee D(\psi) \\
D(\varphi \vee \psi) & := & D(\varphi) \wedge D(\psi) \\
D(\Box\varphi) & := & \Diamond D(\varphi) \\
D(\Diamond\varphi) & := & \Box D(\varphi) \\
D(\mu Z.\varphi(Z)) & := & \nu Z.D(\varphi(\neg Z)) \\
D(\nu Z.\varphi(Z)) & := & \mu Z.D(\varphi(\neg Z))
\end{array}
$$

Observe that the additional negation of the variable Z in the clauses for the fixed point operators ensures that Z occurs positively in φ.

Lemma 5.3.2. For any L_μ-formula φ, any transition system $T = (S, \to, \rho)$ and any valuation $V : Var \longrightarrow \mathcal{P}(S)$ it holds that $[\![D(\varphi)]\!]^T_V = S - [\![\varphi]\!]^T_V$.

Proof. Let $T = (S, \to, \rho)$ be an arbitrary transition system and $V : Var \longrightarrow \mathcal{P}(S)$ an arbitrary valuation. We show that $[\![D(\varphi)]\!]^T_V = S - [\![\varphi]\!]^T_V$ by induction on φ.

▷ Base case:

- Suppose $\varphi = P$. Then $[\![D(P)]\!]^T_V \stackrel{\text{def}}{=} [\![\neg P]\!]^T_V = S - [\![P]\!]^T_V$. The case for $\varphi = Z$ is similar.
- Suppose $\varphi = \neg P$. Then $[\![D(\neg P)]\!]^T_V \stackrel{\text{def}}{=} [\![P]\!]^T_V = S - (S - [\![P]\!]^T_V) = S - [\![\neg P]\!]^T_V$. The case for $\varphi = \neg Z$ is similar.

▷ Induction step:

- Suppose $\varphi = \psi_1 \wedge \psi_2$. Then we have that

$$\begin{aligned}
[\![D(\psi_1 \wedge \psi_2)]\!]^T_V &\stackrel{\text{def}}{=} [\![D(\psi_1) \vee D(\psi_2)]\!]^T_V \\
&= [\![D(\psi_1)]\!]^T_V \cup [\![D(\psi_2)]\!]^T_V \\
&\stackrel{\text{IH.}}{=} (S - [\![\psi_1]\!]^T_V) \cup (S - [\![\psi_2]\!]^T_V) \\
&= S - ([\![\psi_1]\!]^T_V \cap [\![\psi_2]\!]^T_V) \\
&= S - [\![\psi_1 \wedge \psi_2]\!]^T_V
\end{aligned}$$

  The case for $\varphi = \psi_1 \vee \psi_2$ is similar.

- Suppose $\varphi = \Box\psi$. Then we have that

$$\begin{aligned}
[\![D(\Box\psi)]\!]^T_V &\stackrel{\text{def}}{=} [\![\Diamond D(\psi)]\!]^T_V \\
&= \{s \in S | \exists t \in S(s \to t \wedge t \in [\![D(\psi)]\!]^T_V)\} \\
&\stackrel{\text{IH.}}{=} \{s \in S | \exists t \in S(s \to t \wedge t \in (S - [\![\psi]\!]^T_V)\} \\
&= \{s \in S | \exists t \in S(s \to t \wedge t \notin [\![\psi]\!]^T_V)\} \\
&= \{s \in S | s \notin [\![\Box\psi]\!]^T_V)\} \\
&= S - [\![\Box\psi]\!]^T_V
\end{aligned}$$

  The case for $\varphi = \Diamond\psi$ is similar.

– Suppose $\varphi = \mu Z.\psi(Z)$. Then we have that

$$\begin{aligned}
s \in [\![D(\mu Z.\psi(Z))]\!]^T_V &\overset{\text{def}}{\Leftrightarrow} s \in [\![\nu Z.D(\psi(\neg Z))]\!]^T_V \\
&\Leftrightarrow s \in \bigcup\{U \subseteq S | U \subseteq [\![D(\psi(\neg Z))]\!]^T_{V[Z\mapsto U]}\} \\
&\Leftrightarrow \exists U \subseteq [\![D(\psi(\neg Z))]\!]^T_{V[Z\mapsto U]} \text{ and } s \in U \\
&\overset{\text{I.H.}}{\Leftrightarrow} \exists U \subseteq S - [\![\psi(\neg Z)]\!]^T_{V[Z\mapsto U]} \text{ and } s \in U \\
&\Leftrightarrow^* \exists U' \subseteq S \text{ such that } [\![\psi(Z))]\!]^T_{V[Z\mapsto U']} \subseteq U' \text{ and } s \notin U' \\
&\Leftrightarrow s \notin \bigcap\{U' \subseteq S | [\![\psi(Z))]\!]^T_{V[Z\mapsto U']} \subseteq U'\} \\
&\Leftrightarrow s \notin [\![\mu Z.\psi(Z)]\!]^T_V \\
&\Leftrightarrow s \in (S - [\![\mu Z.\psi(Z)]\!]^T_V)
\end{aligned}$$

Notice that the induction hypothesis can be used as it ranges over arbitrary valuations. Moreover, the step also includes an application of the substitution principle (which is needed as the induction hypothesis does technically not range over $\psi(\neg Z)$). For the equivalence labelled by $*$ consider the set $U' = S - U$ where U is the witness of the statement one line above. The case for $\varphi = \nu Z.\psi(Z)$ is similar. □

In the rest of the subsection we define some useful notation which simplifies later proofs.

Definition 5.3.3. Let $t = (V, \rightarrow, \lambda)$ be a pre-proof and $\mathbb{P} = \mathbb{P}(0)\mathbb{P}(1)\mathbb{P}(2)...$ a (possibly infinite) path through t. $\mathbb{P}$ is called a *valid path* if and only if one of the following two conditions holds:

1. $\mathbb{P}$ is a finite path and ends in a axiomatic leaf.

2. $\mathbb{P}$ is an infinite path and there is a ν-trace through $\mathbb{P}$.

A path which is not valid is called *invalid*.

Lemma 5.3.4. A pre-proof $t = (V, \rightarrow, \lambda)$ is a proof if and only if every path through t is valid.

Proof. Trivial. □

Notice that the contraposition of this lemma states that a pre-proof t is not a proof if and only if there exists an invalid path through t.

Corollary 5.3.5. If Γ is not **DT**-provable, then every pre-proof for Γ has an invalid path.

We introduce a corresponding notion of a satisfying path through a pre-tableau:

Definition 5.3.6. Let $t = (V, \rightarrow, \lambda)$ be a pre-tableau and $\mathbb{P} = \mathbb{P}(0)\mathbb{P}(1)\mathbb{P}(2)...$ a (possibly infinite) path through t. $\mathbb{P}$ is called a *satisfying path* if and only if one of the following conditions hold:

1. $\mathbb{P}$ is a finite path which ends in a leaf labelled by $\Box\Delta, \Theta$ where $\Theta \subseteq Lit$ is consistent.

2. $\mathbb{P}$ is an infinite path and every infinite trace through $\mathbb{P}$ is a ν-trace.

A path which is not satisfying is called *unsatisfying*.

Lemma 5.3.7. A pre-tableau $t = (V, \rightarrow, \lambda)$ is a tableau if and only if every path through t is satisfying.

Proof. Suppose $t = (V, \rightarrow, \lambda)$ is a tableau and $\mathbb{P}$ an arbitrary path through t. First suppose $\mathbb{P}$ is a finite path. Then since t is a tableau, the path $\mathbb{P}$ ends in a leaf of t which is labelled by $\Box\Delta, \Theta$ for $\Theta \subseteq Lit$ consistent. This implies that $\mathbb{P}$ is satisfying. If $\mathbb{P}$ is an infinite path, then since every infinite trace is a ν-trace, in particular every infinite trace through $\mathbb{P}$ is a ν-trace, which implies that $\mathbb{P}$ is satisfying. Therefore every path is satisfying. For the other direction suppose $t = (V, \rightarrow, \lambda)$ is a pre-tableau such that every path $\mathbb{P}$ through t is satisfying. Suppose $u \in V$ is a leaf. Then the path $\mathbb{P}$ starting at the root of t and ending at u is finite and therefore u is labelled by $\Box\Delta, \Theta$ for $\Theta \subseteq Lit$ consistent. Suppose tr is an infinite trace in t. As traces are defined relative to paths, tr is an infinite trace through some path $\mathbb{P}$, which implies that $\mathbb{P}$ is infinite. This in turn implies that tr is a ν-trace. Together we conclude that t is a tableau. □

The next concept introduced is that of corresponding paths which links paths through pre-proofs with paths through pre-tableaux. We first need the following definition:

Definition 5.3.8. Let φ be an L_μ-formula. The formula $\overline{\varphi}$ is defined to be φ where each negated occurrence of a fixed point variable $\neg Y$ in φ is replaced by Y.

Given a sequent $\Gamma = \{\varphi_1, ..., \varphi_n\}$ we denote by $\overline{\Gamma}$ the set $\{\overline{\varphi_1}, ..., \overline{\varphi_n}\}$.

Definition 5.3.9. Let $t = (V, \rightarrow, \lambda)$ be a pre-proof and $t' = (V', \rightarrow', \lambda')$ be a pre-tableau. Let $\mathbb{P}$ be a path through t and $\mathbb{P}'$ a path through t'. We call $\mathbb{P}$ and $\mathbb{P}'$ *corresponding paths* if and only if the following two conditions hold:

1. Either both $\mathbb{P}$ and $\mathbb{P}'$ are infinite paths or they are both finite and have the same length.

2. For any $i \in \omega$ such that $\mathbb{P}(i)$ and $\mathbb{P}'(i)$ exist:

$$\overline{D(\lambda(\mathbb{P}(i)))} = \lambda'(\mathbb{P}'(i))$$

A few comments about this definition:

- ▷ As we are only considering pre-proofs and pre-tableaux for closed formulas, notice that the root of such a tree is labelled by Γ such that $\overline{\Gamma} = \Gamma$.

- ▷ The general intuition is that if two paths are corresponding, they are identical modulo dualism. This means that if they are finite, they have the same length and whenever on one path a rule $(*)$ is applied to a formula φ, then on the other path its dual tableaux rule is applied to $D(\varphi)$. The dual rules are thereby given according to the translation D. That is, the dual tableaux rules of $(\wedge)$ are $(\vee)_i$, the dual rule of $(\vee)$ is $(\wedge)$, the dual rule of $(\Box)$ is (mod) and so on.

- ▷ The additional translation given by the over-line that replaces negated variables by non-negated variables is necessary due to the formulation of the fixed point rules. Given a formula $\sigma Z.\varphi(Z)$, after applying the fixed point rule (σ), the variable Z occurs freely in the rest of the prooftree and D would map Z onto its negation whenever it occurs as a subformula. This is clearly not the intended meaning, as such variables should still be considered to be bound and so we delete the negation in such occurrences. Another solution would be to replace the fixed point rules by the following rule used for example in [18]

$$\frac{\Gamma, \varphi(\sigma Z.\varphi(Z))}{\Gamma, \sigma Z.\varphi(Z)}\ (\sigma)$$

  where $\sigma \in \{\mu, \nu\}$. This rule combines our fixed point rule with our variable rule. Using this rule implies that no free variables occur in the prooftree. The rule above and our fixed point rules are equivalent.

Example 5.3.10. Let $\varphi = \nu Y.\Diamond Y \wedge \Box(\neg P \vee \neg Q)$ and $D(\varphi) = \mu Y.\Box Y \vee \Diamond(P \wedge Q)$. The following is a pre-proof for φ:

$$\cfrac{\cfrac{\cfrac{\Diamond Y \qquad \cfrac{\cfrac{\neg P, \neg Q}{\neg P \vee \neg Q}\ (\vee)}{\Box(\neg P \vee \neg Q)}\ (\Box)}{\Diamond Y \wedge \Box(\neg P \vee \neg Q)}\ (\wedge)}{Y}\ (Y)}{\nu Y.\Diamond Y \wedge \Box(\neg P \vee \neg Q)}\ (\nu)$$

Below is a tableau for $D(\varphi)$:

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{P, Q}{P \wedge Q}\ (\wedge)}{\Diamond(P \wedge Q)}\ (\text{mod})}{\Box Y \vee \Diamond(P \wedge Q)}\ (\vee)}{Y}\ (\text{Y})}{\mu Y.\Box Y \vee \Diamond(P \wedge Q)}\ (\mu)$$

The pre-proof contains two finite branches both leading to a leaf which is not axiomatic. The tableau has only a single finite branch. Notice that the right branch of the pre-proof and the branch of the tableau are corresponding; they have the same length and whenever a node in the right branch of the pre-proof is labelled by Γ, then its corresponding node in the tableau is labelled by $\overline{D(\Gamma)}$. If we choose the left disjunct $\Box Y$ at the rule $(\vee)$ in the tableau, we obtain a different tableau for $D(\varphi)$ whose only branch corresponds to the left branch of the pre-proof.

Lemma 5.3.11. Let $t = (V, \rightarrow, \lambda)$ be a pre-proof for Γ and $t' = (V', \rightarrow', \lambda')$ a pre-tableau for Γ'. Let $\mathbb{P}$ be a path through t and $\mathbb{P}'$ a path through t' such that $\mathbb{P}$ and $\mathbb{P}'$ are corresponding. Then $\mathbb{P}$ is invalid if and only if $\mathbb{P}'$ is satisfying.

Proof. Suppose $\mathbb{P}$ is invalid. We distinguish two cases:

1. Suppose $\mathbb{P} = \mathbb{P}(0)\mathbb{P}(1)...\mathbb{P}(n)$ is a finite path. Since $\mathbb{P}$ is invalid, the leaf $\mathbb{P}(n)$ is not axiomatic, which implies that $\lambda(\mathbb{P}(n)) = \Diamond\Sigma, \Theta$ where $\Theta \subseteq \mathit{Lit}$ is consistent. Since $\mathbb{P}$ and $\mathbb{P}'$ are corresponding we have that $\mathbb{P}' = \mathbb{P}'(0)\mathbb{P}'(1)...\mathbb{P}'(n)$ and

$$\lambda(\mathbb{P}'(n)) = \overline{D(\lambda(\mathbb{P}(n))} = \Box\overline{D(\Sigma)}, \overline{D(\Theta)}$$

   Notice that since $\Theta \subseteq \mathit{Lit}$ is consistent also $\overline{D(\Theta)} = D(\Theta) \subseteq \mathit{Lit}$ is consistent. Therefore, $\mathbb{P}'(n)$ is labelled by boxed formulas and consistent literals, which implies that $\mathbb{P}'$ is satisfying.

2. Suppose $\mathbb{P}$ is an infinite path. Then $\mathbb{P}'$ is infinite as well. Since $\mathbb{P}$ is invalid, there is no infinite ν-trace through $\mathbb{P}$. By Lemma 4.2.3 every infinite path contains an infinite trace. Hence there exists an infinite trace through $\mathbb{P}$ and every such infinite trace through $\mathbb{P}$ is a μ-trace. By definition of the translation $D(\cdot)$ every infinite trace through $\mathbb{P}'$ is a therefore a ν-trace, which implies that $\mathbb{P}'$ is satisfying.

For the other direction suppose that $\mathbb{P}'$ is satisfying. Again we distinguish two cases:

1. Suppose $\mathbb{P}' = \mathbb{P}'(0)\mathbb{P}'(1)...\mathbb{P}'(n)$ is a finite path. Since $\mathbb{P}'$ is satisfying, the leaf $\mathbb{P}'(n)$ is labelled by $\Box\overline{D(\Delta)}, \overline{D(\Theta)}$ where $\overline{D(\Theta)} \subseteq \mathit{Lit}$ is consistent. Since $\mathbb{P}$ and $\mathbb{P}'$ are corresponding we have that $\mathbb{P} = \mathbb{P}(0)\mathbb{P}(1)...\mathbb{P}(n)$ and

$$\lambda(\mathbb{P}(n)) = \Diamond\Delta, \Theta$$

   where $\Theta \subseteq \mathit{Lit}$ is consistent. Therefore $\mathbb{P}(n)$ is not an axiomatic leaf, which implies that $\mathbb{P}$ is invalid.

2. Suppose $\mathbb{P}'$ is an infinite path. Then $\mathbb{P}$ is infinite as well. Furthermore since $\mathbb{P}'$ is satisfying every infinite trace through $\mathbb{P}'$ is a ν-trace. This implies that every trace through $\mathbb{P}$ is a μ-trace and so that there is no infinite ν-trace through $\mathbb{P}$. Thus $\mathbb{P}$ is invalid. □

5.3.2 Completeness of DT

For completeness of **DT** we establish the following connection between **DT** and **T**:

If a sequent Γ does not have a **DT**-proof, then $D(\Gamma)$ has a tableau. (5.1)

Completeness of **DT** is then derived from this result and soundness of **T**. In order to prove (5.1), we show how to build a tableau for $D(\Gamma)$ using only the information that Γ has no proof. Starting from an arbitrary pre-proof for Γ, the tableau is built in such a way, that every path corresponds to a path through the pre-proof. The only difficulty is the existence of the rule (*mod*). Suppose we reach a node in the tableau labelled by

$$\Box\overline{D(\varphi_1)}, ..., \Box\overline{D(\varphi_n)}, \Diamond\overline{D(\psi_1)}, ..., \Diamond\overline{D(\psi_k)}, \overline{D(\Theta)}$$

for $\overline{D(\Theta)} \subseteq Lit$ consistent. Applying the rule (*mod*) to such a node generates k children each of them labelled by $\overline{D(\varphi_1)}, ..., \overline{D(\varphi_n)}, \overline{D(\psi_i)}$ for some $1 \leq i \leq k$. The corresponding node in the pre-proof is labelled by

$$\Diamond\varphi_1, ..., \Diamond\varphi_n, \Box\psi_1, ..., \Box\psi_k, \Theta$$

where $\Theta \subseteq Lit$ is consistent. Applying the dual rule ($\Box$) only generates one child labelled by $\varphi_1, ..., \varphi_n, \psi_i$ for some $1 \leq i \leq k$ as we have to choose which boxed formula survives. Hence, only one of the k-paths generated by the application of (*mod*) corresponds to the path in the pre-proof. Having only a single pre-proof at hand does therefore not suffice to build the tableau. Luckily, since Γ has no proof, we can use every possible pre-proof to build the tableau. However, we are only interested in those pre-proofs that are identical to the chosen pre-proof up to the node where the box-rule is applied.

Definition 5.3.12. Suppose $t = (V, \rightarrow, \lambda)$ is a pre-proof and $\mathbb{P}$ is a path through t where for some $n \in \omega$ the node $\mathbb{P}(n)$ is labelled by $\Diamond\Delta, \Box\varphi_1, ..., \Box\varphi_k, \Theta$ for $\Theta \subseteq Lit$ consistent, $k \geq 2$ and $\mathbb{P}(n+1)$ is labelled by Δ, φ_i. A pre-proof $t' = (V', \rightarrow', \lambda')$ is called *quasi-identical to t with respect to* $\mathbb{P}$, if t' is everywhere identical to t but in the subtree given by $Up(\mathbb{P}(n))$.

That is, two pre-proofs are quasi-identical if they only differ in one path, where different boxed formulas are chosen to survive a ($\Box$)-rule. Notice that in the definition the pre-proof t and the path $\mathbb{P}$ are fixed, but not the specific node $\mathbb{P}(n)$. That is, if there are two suitable nodes $\mathbb{P}(n)$ and $\mathbb{P}(m)$ for $m \neq n$, then there exist quasi-identical pre-proofs to t with respect to $\mathbb{P}$ which differ from t at node $\mathbb{P}(n)$ and others which differ at node $\mathbb{P}(m)$. They are both counted as quasi-identical. In case no suitable node in $\mathbb{P}$ exist, we say that t has no quasi-identical pre-proofs with respect to $\mathbb{P}$. In order to build the tableau for $D(\Gamma)$, it suffices to look at an arbitrary pre-proof for Γ and every quasi-identical pre-proof with respect to a specific path. The specific path is required to be completely invalid.

Definition 5.3.13. Let $t = (V, \rightarrow, \lambda)$ be a pre-proof for Γ and $\mathbb{P}$ a path through t. The path $\mathbb{P}$ is *completely invalid* if and only if $\mathbb{P}$ is invalid and if $t' = (V', \rightarrow', \lambda')$ is a quasi-identical pre-proof to t with respect to $\mathbb{P}$ which differs at node $\mathbb{P}(n)$, then there exists a path $\mathbb{P}'$ through t' such that $\mathbb{P}'(0)...\mathbb{P}'(n) = \mathbb{P}(0)...\mathbb{P}(n)$ and $\mathbb{P}'$ is invalid.

Notice that in case there are no quasi-identical pre-proofs to t with respect to $\mathbb{P}$, then $\mathbb{P}$ is completely invalid if and only if $\mathbb{P}$ is invalid. We prove two lemmas about completely invalid paths. First, we show that given a sequent which is not provable, every pre-proof for that sequent contains a completely invalid path. Second, we show that the invalid path $\mathbb{P}'$ through the quasi-identical pre-proof is itself completely invalid.

Lemma 5.3.14. Let Γ be a sequent and suppose that Γ does not have a proof. Then every pre-proof for Γ has a completely invalid path.

Proof. Suppose that Γ does not have a proof. By corollary 5.3.5 every pre-proof of Γ has an invalid path. Suppose towards a contradiction that there exists a pre-proof $t = (V, \rightarrow, \lambda)$ for Γ which does not have a completely invalid path. Thus for every invalid path $\mathbb{P}$ there is a quasi-identical pre-proof t' to t with respect to $\mathbb{P}$ which differs at node (say) $\mathbb{P}(n)$ such that every path through t' which is identical to $\mathbb{P}$ up to the node $\mathbb{P}(n)$ is valid. We then replace the sub-tree of t rooted in $\mathbb{P}(n)$ by the sub-tree of t' rooted in $\mathbb{P}(n)$ (i.e. we turn t into t') and so eliminate the invalid path $\mathbb{P}$ without creating any new invalid paths. By iterating the procedure we eliminate one by one every invalid path through t until we are left with a pre-proof t^P for Γ in which every path is valid. By lemma 5.3.4 this implies that t^P is a proof contradicting our assumption that Γ does not have a proof. We conclude that every pre-proof for Γ has a completely invalid path. □

Lemma 5.3.15. Suppose $t = (V, \rightarrow, \lambda)$ is a pre-proof for Γ and $\mathbb{P}$ is a completely invalid path through t. Let $t' = (V', \rightarrow', \lambda')$ be a quasi-identical pre-proof to t with respect to $\mathbb{P}$ which differs at $\mathbb{P}(n)$. There exists a completely invalid path in t' which passes through the node $\mathbb{P}(n)$.

Proof. Suppose towards a contradiction that there is no completely invalid path in t' which passes through $\mathbb{P}(n)$. Let $\mathbb{P}'$ be an invalid path in t' passing through $\mathbb{P}(n)$. As $\mathbb{P}'$ is not completely invalid, there exists a quasi-identical pre-proof t'' to t' with respect to $\mathbb{P}'$ which differs at $\mathbb{P}'(m)$ such that every path in t'' passing through $\mathbb{P}'(m)$ is valid. Notice that $m > n$ as otherwise it would contradict the assumption of $\mathbb{P}$ being completely invalid. We replace the sub-tree of t' rooted at $\mathbb{P}'(m)$ by the sub-tree of t'' rooted at $\mathbb{P}'(m)$ and so eliminate one invalid path passing through $\mathbb{P}(n)$ without introducing new invalid paths. By iterating the procedure we therefore replace step by step every invalid path in t' passing through $\mathbb{P}(n)$ by valid paths until we are left with a pre-proof t^P which is a quasi-identical to t with respect to $\mathbb{P}$ and every path in t^P passing through $\mathbb{P}(n)$ is valid, contradicting the assumption that $\mathbb{P}$ is completely invalid. We conclude that there exists a completely invalid path in t passing through $\mathbb{P}(n)$. □

We are now ready to prove the main theorem of this subsection.

Theorem 5.3.16. Let Γ be a sequent. If Γ does not have a proof, then $D(\Gamma)$ has a tableau.

Proof. Suppose Γ does not have a proof. Let $t_1 = (V_1, \rightarrow_1, \lambda_1)$ be a pre-proof for Γ. By lemma 5.3.14 t_1 has a completely invalid path $\mathbb{P}_1$. Let $\{t_i | i \in I\}$ be the collection of quasi-identical pre-proofs to t_1 with respect to $\mathbb{P}_1$. The set I is some index set which might be the singleton $\{1\}$ in case the completely invalid path of t_1 does not have any occurrences of the ($\Box$)-rule with more than one boxed formula to be chosen. By lemma 5.3.15 each pre-proof t_i for $i \in I$ has a completely invalid path $\mathbb{P}_i$ which is identical to $\mathbb{P}_1$ up to some node $\mathbb{P}_1(n_i)$ where the ($\Box$)-rule is applied.

We show how to construct a pre-tableau $t = (V, \rightarrow, \lambda)$ for $D(\Gamma)$.

- ▷ Let the root r_t of t be labelled by $D(\Gamma)$. Notice that $\mathbb{P}_i(0) = r_t^i$ and $\mathbb{P}(0) = r_t$ are corresponding initial segments of paths for each $i \in I$, where r_t^i is the root of the pre-proof t_i.
- ▷ Suppose we have constructed t up to the node u where $u = \mathbb{P}(n)$ for some initial segment of a path $\mathbb{P}$ and $\mathbb{P}(0)...\mathbb{P}(n)$ corresponds to the initial segment $\mathbb{P}_i(0)...\mathbb{P}_i(n)$ of some completely invalid path $\mathbb{P}_i$. We show how to extend the pre-tableau t:
  - Suppose $\mathbb{P}_i(n)$ is labelled by Δ, φ where $\varphi = \varphi_1 \vee \varphi_2$ and the rule applied is ($\vee$) which generates the node $\mathbb{P}_i(n+1)$ labelled by $\Delta, \varphi_1, \varphi_2$. As the two initial segments are corresponding, $\mathbb{P}(n)$ is labelled by $\overline{D(\Delta)}, \overline{D(\varphi)}$ where $\overline{D(\varphi)} = \overline{D(\varphi_1)} \wedge \overline{D(\varphi_2)}$. Therefore we apply the rule ($\wedge$) to generate the node $\mathbb{P}(n+1)$ which is labelled by $\overline{D(\Delta)}, \overline{D(\varphi_1)}, \overline{D(\varphi_2)}$. By construction the initial segments $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1)$ and $\mathbb{P}_i(0)...\mathbb{P}_i(n)\mathbb{P}_i(n+1)$ are corresponding.
  - Suppose $\mathbb{P}_i(n)$ is labelled by Δ, φ where $\varphi = \varphi_1 \wedge \varphi_2$ and the rule applied is ($\wedge$) to generate the nodes v and w labelled by Δ, φ_1 and Δ, φ_2 respectively. Suppose without loss of generality that $\mathbb{P}_i(n+1) = v$. By assumption $\mathbb{P}(n)$ is labelled by $\overline{D(\Delta)}, \overline{D(\varphi)}$, where $\overline{D(\varphi)} = \overline{D(\varphi_1)} \vee \overline{D(\varphi_2)}$. Then we apply the rule ($\vee$) to generate the node $\mathbb{P}(n+1)$ labelled by $\overline{D(\Delta)}, \overline{D(\varphi_1)}$. Observe that the initial segments $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1)$ and $\mathbb{P}_i(0)...\mathbb{P}_i(n)\mathbb{P}_i(n+1)$ are corresponding.
  - Suppose $\mathbb{P}_i(n)$ is labelled by $\Diamond\Delta, \Box\varphi_1, ..., \Box\varphi_k, \Theta$ where $\Theta \subseteq Lit$ is consistent, $k \geq 1$ and the rule applied is ($\Box$) which generates the node $\mathbb{P}_i(n+1)$ which is labelled by Δ, φ_i for some $1 \leq i \leq k$. By assumption the node $\mathbb{P}(n)$ is labelled by

    $$\overline{D(\Diamond\Delta)}, \overline{D(\Box\varphi_1)}, ..., \overline{D(\Box\varphi_k)}, \overline{D(\Theta)}$$

    which is

    $$\Box\overline{D(\Delta)}, \Diamond\overline{D(\varphi_1)}, ..., \Diamond\overline{D(\varphi_k)}, \overline{D(\Theta)}$$

    Notice that $\overline{D(\Theta)} = D(\Theta) \subseteq Lit$ is consistent. Therefore we apply the rule (mod) to generate nodes $v_1, ..., v_k$ labelled by $\overline{D(\Delta)}, \overline{D(\varphi_i)}$ for $1 \leq i \leq k$ respectively. First of all notice that the initial segments $\mathbb{P}(0)...\mathbb{P}(n)v_i$ and $\mathbb{P}_i(0)...\mathbb{P}_i(n)\mathbb{P}_i(n+1)$

are corresponding. Next let $j \neq i$. Consider a pre-proof t' for Γ which is quasi-identical to t_i with respect to $\mathbb{P}_i$ and differs above the node $\mathbb{P}_i(n)$, where at the application of $(\Box)$ at node $\mathbb{P}_i(n)$ the successor is labelled by the sequent Δ, φ_j. If $t_i = t_1$, then t' is quasi-identical to t_1 with respect to $\mathbb{P}_1$. If $t_i \neq t_1$, then since t_i is quasi-identical to t_1 with respect to $\mathbb{P}_1$ the path $\mathbb{P}_i$ is identical to $\mathbb{P}_1$ in the first m steps and $m < n$ (otherwise t_i can be considered to be t_1). This implies that the path through t' is also identical to the path $\mathbb{P}_1$ in the first m steps and t' is therefore also quasi-identical to t_1 with respect to $\mathbb{P}_1$. Thus in both cases $t' = t_l$ for some $l \in I$. Moreover, the sequence $\mathbb{P}_i(0)...\mathbb{P}_i(n)v$ of nodes in t_l where v is the successor of $\mathbb{P}_i(n)$ is an initial segment of the completely invalid path $\mathbb{P}_l$. Lastly, $\mathbb{P}_l(0)...\mathbb{P}_l(n)\mathbb{P}_l(n+1)$ corresponds to $\mathbb{P}(0)...\mathbb{P}(n)v_j$. As $j \neq i$ was arbitrary we have that each of $v_1, ..., v_k$ extends a path corresponding to a completely invalid path.

- Suppose $\mathbb{P}_i(n)$ is labelled by $\Delta, \mu Z.\varphi(Z)$ or $\Delta, \nu Z.\varphi(Z)$ and the rule applied is (μ) or (ν) respectively and generates the node $\mathbb{P}_i(n+1)$ which is labelled by Z. By assumption $\mathbb{P}(n)$ is labelled by $\overline{D(\Delta)}, \overline{D(\mu Z.\varphi(Z))}$ where $\overline{D(\mu Z.\varphi(Z))} = \nu Z.\overline{D(\varphi(\neg Z))}$ or by $\overline{D(\Delta)}, \overline{D(\nu Z.\varphi(Z))}$ where $\overline{D(\nu Z.\varphi(Z))} = \mu Z.\overline{D(\varphi(\neg Z))}$. Then we apply the rule (ν) or (μ) respectively to generate the node $\mathbb{P}(n+1)$ labelled by $\overline{D(\Delta)}, Z$. Notice that $D(Z) = \neg Z$ and thus $\overline{D(Z)} = Z$. Hence the initial segments of paths $\mathbb{P}_i(0)...\mathbb{P}_i(n)\mathbb{P}_i(n+1)$ and $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1)$ are corresponding.
- Suppose $\mathbb{P}_i(n)$ is labelled by Δ, Z where Z identifies $\varphi(Z)$ and the rule applied is (Z) generating the node $\mathbb{P}_i(n+1)$ labelled by $\Delta, \varphi(Z)$. By assumption $\mathbb{P}(n)$ is labelled by $\overline{D(\Delta)}, Z$ where Z identifies $\overline{D(\varphi(Z))}$. Thus we apply the rule (Z) to generate the node $\mathbb{P}(n+1)$ labelled by $\overline{D(\Delta)}, \overline{D(\varphi(Z))}$ and thus the initial segments $\mathbb{P}_i(0)...\mathbb{P}_i(n)\mathbb{P}_i(n+1)$ and $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1)$ are corresponding.

This finishes the construction of the tree t. Notice that t is a finite branching tree whose root is labelled by $D(\Gamma)$ and which is generated by the tableaux-rules. Moreover, by construction every path in t corresponds to a completely invalid path in a pre-proof of Γ. Since every completely invalid path is also invalid, lemma 5.3.11 implies that every path through t is satisfying. Therefore lemma 5.3.7 implies that t is a tableau for $D(\Gamma)$. $\Box$

Theorem 5.3.17 (Completeness of **DT**)**.** If a sequent Γ is valid, then $\mathbf{DT} \vdash \Gamma$.

Proof. We proceed by contraposition. Suppose a sequent Γ does not have a **DT**-proof. By the previous theorem $D(\Gamma)$ has a tableau. The soundness result for **T** implies that $\bigwedge D(\Gamma)$ is satisfiable, which by basic propositional reasoning and lemma 5.3.2 implies that $\bigvee \Gamma$ is not valid. We conclude that **DT** is complete. $\Box$

We have therefore established that **DT** is complete. We move on to prove soundness.

5.3.3 Soundness of DT

For establishing soundness of **DT**, we prove the converse of (5.1):

$$\text{If } D(\Gamma) \text{ has a tableau, then } \Gamma \text{ does not have a proof} \tag{5.2}$$

Soundness of **DT** is then derived from (5.2) and completeness of **T**. In order to prove (5.2), we show that given a tableau for $D(\Gamma)$, every pre-proof of Γ has an invalid path. For that, we again use the notion of corresponding paths introduced in the previous subsection. It suffices to show that every pre-proof of Γ contains a path that is corresponding to some path in the tableau for $D(\Gamma)$. Unfortunately, this is in general not true. There are simply too many possibilities of pre-proofs for Γ, as that every single one of them has such a path. However, it is true if we restrict to a specific form of tableaux and proofs, namely those in normal form.

Definition 5.3.18. Let Γ be a sequent and let $Sub(\Gamma)$ be the set of all subformulas of formulas in Γ. Suppose $|Sub(\Gamma)| = k$. Let $e : Sub(\Gamma) \longrightarrow \{1, ..., k\}$ be an enumeration of $Sub(\Gamma)$.

- ▷ A *pre-tableau in normal form with respect to* e for Γ is a pre-tableau $t = (V, \rightarrow, \lambda)$ for Γ such that whenever there is a node $u \in V$ labelled by $\Delta \subseteq Sub(\Gamma)$, the rule $(*)$ applied at u is applied to the least formula $\varphi \in \Delta$ in the enumeration for which a rule can be applied. A *tableau in normal form with respect to* e is a tableau which is a pre-tableau in normal form with respect to e.
- ▷ Let u be a node in a tableau and let φ be the least formula labelling u to which a rule can be applied. If the rule $(*)$ applied at u is not applied to φ, then u is called a *non-normal* node.

A formula to which no rule can be applied is either a literal or it is a box- or diamond-formula, such that the sequent labelling the current node is not suitable for an application of (mod). Pre-tableaux for Γ in normal form are defined relative to an enumeration of the subformulas of Γ. The enumeration dictates exactly which rules have to be applied in what order to which formulas. However, pre-tableaux in normal form with respect to an enumeration e are not unique. Observe that applying a rule to a formula such as $\varphi_1 \vee \varphi_2$ in a normal form tableau does not imply that we have to choose the smaller one of φ_1 and φ_2 with respect to e. Indeed we are allowed to choose the larger formula as well, as long as the rule applied is always applied to the least formula according to e to which a formula can be applied. That is, there might exits different pre-tableaux in normal form with respect to e for a given sequent. In case some node is labelled by $\Box\Gamma, \Diamond\{\varphi_1, ..., \varphi_n\}, \Theta$ for $\Theta \subseteq Lit$ consistent, the only rule that can be applied is (mod) and (mod) is considered to be applied to every formula labelling the node. Therefore such a node is trivially normal.

Lemma 5.3.19. Let Γ be a sequent and e an arbitrary enumeration of $Sub(\Gamma)$. If Γ has a tableau, then Γ has a tableau in normal form with respect to e.

Using an arbitrary enumeration encodes the idea that it does not matter in what order rules are applied to formulas in tableaux between two applications of (mod).

Proof. Suppose $t = (V, \to, \lambda)$ is a tableau for Γ which is not in normal form with respect to the enumeration e. Let u be the lowermost non-normal node in t; that is, $u \in V$ is such that the branch up to u is in normal form, but at u the rule applied is not applied to the least formula according to the enumeration. Suppose that u is labelled by $\varphi_1, ..., \varphi_n$ and without loss of generality that $e(\varphi_1) < e(\varphi_2) < ... < e(\varphi_n)$. Moreover suppose without loss of generality that a rule can be applied to φ_1. First of all notice that there exists a node v in the tableau above u at which a rule is applied to φ_1, as otherwise t would not be a tableau. We assume that v is the first node after u at which a rule is applied to φ_1. Let $(*)_1, ..., (*)_k$ be the rules applied between the nodes u and v to formulas other than φ_1. Notice that none of these rules can be an instance of (mod), as this would contradict the assumption that the rule is not applied to φ_1. Therefore there is no branching in the path between u and v. Furthermore, the rule $(*)$ applied at node v to generate its successor node w cannot be (mod) either, as this would imply that either u was already labelled by diamond- and boxed-formulas and literals only, contradicting the assumption that there was a rule applied at u to some other formula than φ_1, or it was not possible to apply a rule to φ_1 at node u, contradicting this exact assumption. Therefore by first applying the rule $(*)$ at node u to φ_1 and then the rules $(*)_1, ..., (*)_k$ we obtain the same successor node w of v labelled by the same formulas as in the tableau t. Thus we have demonstrated that we can eliminate the lowermost non-normal node in t without damaging t being a tableau. Therefore by working from the root of the tableau t upwards we can one by one eliminate every non-normal node. Notice that while eliminating a non-normal node might create a new non-normal node, this newly created non-normal node occurs higher up in the tableau and is thus not problematic, as it will be eliminated just a few steps later. Therefore we can turn t into a tableau in normal form with respect to e. □

We introduce a similar notion for pre-proofs.

Definition 5.3.20. Let Γ be a sequent and e an enumeration of $Sub(\Gamma)$. A *pre-proof in normal form with respect to e* for Γ is a pre-proof $t = (V, \to, \lambda)$ for Γ such that whenever there is a node $u \in V$ which is labelled by $\Delta \subseteq Sub(\Gamma)$, the rule $(*)$ applied at u is applied to the least formula in Δ according to the enumeration e, for which a rule can be applied. A *proof in normal form with respect to e* is a proof which is a pre-proof in normal form with respect to e.

Lemma 5.3.21. Let Γ be a sequent and e an arbitrary enumeration of $Sub(\Gamma)$. If Γ has a proof, then Γ has a proof in normal form with respect to e.

Proof. Suppose $t = (V, \to, \lambda)$ is a proof for Γ which is not in normal form with respect to the enumeration e. Let u be the lowermost non-normal node in a branch of t. That is, $u \in V$ is such that the branch up to u is in normal form, but the rule applied at node u is not applied to the least formula labelling u according to e for which a rule could be applied. Suppose that u is labelled by $\varphi_1, ..., \varphi_n$ and without loss of generality $e(\varphi_1) < e(\varphi_2) < ... < e(\varphi_n)$. Furthermore, suppose without loss of generality that a rule $(*)$ can be applied to φ_1. We distinguish two cases:

- ▷ Suppose no rule is applied to φ_1 at any higher node than u. This implies that every path going through u is finite (as an infinite path can only occur when the rule $(\Box)$ is applied as every formula is guarded, which implies that there is a node above u where a rule is applied to φ_1). Since t is a proof, every path through u ends in an axiomatic leaf labelled by $\Sigma, P, \neg P$ for some $P \in Prop$. Notice that φ_1 does not have a modal operator as main connective, as this would contradict either the assumption that a rule can be applied to φ_1 at u or the assumption that the rule applied at u is not applied to φ_1. Therefore we can apply the rule $(*)$ to φ_1 at node u and then apply the same rules to the same formulas in the same order as before which implies that every path through u still ends in an axiomatic leaf, now labelled by $\Delta, \Sigma, P, \neg P$ where $\Delta \subseteq Sub(\varphi_1)$. Notice that in case the main connective of φ_1 is a conjunction and the rule applied is $(\wedge)$, every path which previously passed through u splits into two paths which both lead into an axiomatic leaf.
- ▷ Suppose there exists a later node where the rule $(*)$ is applied to φ_1. By the previous point we can assume without loss of generality that whenever t branches above u in each branch there occurs an application of $(*)$ to φ_1 eventually. So suppose there exist $v_1, ..., v_k$ above u such that in each of these nodes the rule $(*)$ is applied to φ_1 (notice that there exists only one rule which can be applied to φ_1, namely the rule that works on φ_1's main connective). Once again the main connective of φ_1 cannot be a modal operator, as this contradicts either the assumption that a rule could be applied to φ_1 at u or the assumption that no rule is applied to φ_1 at u. Consequently there is no instance of $(\Box)$ among the rules applied in each branch between u and v_i. Similarly, the rule $(*)$ applied to φ_1 at the nodes $v_1, ..., v_k$ is not $(\Box)$. Therefore by first applying the rule $(*)$ at node u and then for each of the previous branches the same rules in the same order leads to the successor nodes $w_1, ..., w_{k'}$ for $k' \geq k$ of $v_1, ..., v_k$ and each of the successor nodes is labelled by the same formulas as previously in the proof. So we can once again eliminate this non-normal node without damaging t of being a proof.

Hence by working from the root upwards we can one by one eliminate every non-normal node in t. While eliminating a non-normal node might create a new non-normal node (or even several), this newly created non-normal nodes occur higher up in the proof and are thus not problematic, as they are eliminated just a few steps later. Therefore Γ has a proof in normal form with respect to e. □

Let Γ be a sequent and let e be an enumeration of $Sub(\Gamma)$. Let $\hat{e}$ denote the enumeration for

$\overline{Sub(D(\Gamma))}$[1] such that for every $\varphi \in \Gamma$:

$$e(\varphi) = n \;\Leftrightarrow\; \hat{e}(\overline{D(\varphi)}) = n$$

Let us now prove the property (5.2) restricted to tableaux and pre-proofs in normal form.

Theorem 5.3.22. Let Γ be a sequent and e an arbitrary enumeration of $Sub(\Gamma)$. If $D(\Gamma)$ has a tableau in normal form with respect to the enumeration $\hat{e}$, then Γ does not have a proof in normal form with respect to e.

Proof. Let Γ be a sequent, e an arbitrary enumeration of $Sub(\Gamma)$ and $t' = (V', \to', \lambda')$ a tableau for $D(\Gamma)$ in normal form with respect to the enumeration $\hat{e}$ of $\overline{Sub(D(\Gamma))}$. Let $t = (V, \to, \lambda)$ be an arbitrary pre-proof for Γ in normal form with respect to e. We first show that there exists a path $\mathbb{P}$ through t and a path $\mathbb{P}'$ through t' which are corresponding, by simultaneously constructing both $\mathbb{P}$ and $\mathbb{P}'$.

- ▷ Let $\mathbb{P}(0) = r_t$ and $\mathbb{P}'(0) = r'_t$. Notice that these initial segments of paths are corresponding.

- ▷ Suppose we have constructed $\mathbb{P}(0)\mathbb{P}(1)...\mathbb{P}(n)$ and $\mathbb{P}'(0)\mathbb{P}'(1)...\mathbb{P}'(n)$ which are corresponding. We show how to extend both paths (assuming that they can be extended):

  - Suppose $\mathbb{P}(n)$ is labelled by Δ where the least formula in Δ according to the enumeration e and for which a rule can be applied is $\varphi_0 \vee \varphi_1$. Since t is a pre-proof for Γ in normal form the rule applied at $\mathbb{P}(n)$ is $(\vee)$ which generates a new node u labelled by
  $$(\Delta - \{\varphi_0 \vee \varphi_1\}), \varphi_0, \varphi_1$$
  By assumption $\mathbb{P}'(n)$ is labelled by $\overline{D(\Delta)}$ and $\overline{D(\varphi_0 \vee \varphi_1)} = \overline{D(\varphi_0)} \wedge \overline{D(\varphi_1)}$ is the least formula with respect to $\hat{e}$. So since t' is a tableau in normal form the rule applied at $\mathbb{P}'(n)$ is $(\wedge)$ which generates the node u' labelled by
  $$(\overline{D(\Delta)} - \{\overline{D(\varphi_0 \vee \varphi_1)}\}), \overline{D(\varphi_0)}, \overline{D(\varphi_1)}$$
  Therefore let $\mathbb{P}(n+1) = u$ and $\mathbb{P}'(n+1) = u'$. By construction $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1)$ and $\mathbb{P}'(0)...\mathbb{P}'(n)\mathbb{P}'(n+1)$ are corresponding.
  - Suppose $\mathbb{P}(n)$ is labelled by Δ where the least formula in Δ according to e is $\varphi_0 \wedge \varphi_1$. Then the rule applied at $\mathbb{P}(n)$ is $(\wedge)$ which generates two new nodes u_0 and u_1 which are labelled by Δ', φ_0 and Δ', φ_1 respectively, where $\Delta' = \Delta - \{\varphi_0 \wedge \varphi_1\}$. By assumption $\mathbb{P}'(n)$ is labelled by $\overline{D(\Delta)}$ and the least formula according to $\hat{e}$ is $\overline{D(\varphi_0 \wedge \varphi_1)} = \overline{D(\varphi_0)} \vee \overline{D(\varphi_1)}$. The rule applied at $\mathbb{P}'(n)$ is therefore $(\vee)_i$ and generates a new node u' which is labelled by $\overline{D(\Delta')}, \overline{D(\varphi_i)}$ for $i \in \{0,1\}$. Let $\mathbb{P}(n+1) = u_i$ and let $\mathbb{P}'(n+1) = u'$. By construction $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1)$ and $\mathbb{P}'(0)...\mathbb{P}'(n)\mathbb{P}'(n+1)$ are corresponding.

[1] Recall that by the definition of subformulas negated atoms are considered to be atomic, which means that they do not have any other subformulas but themselves. Consequently $|Sub(\Gamma)| = |\overline{Sub(D(\Gamma))}|$.

– Suppose $\mathbb{P}(n)$ is labelled by $\Diamond\Delta, \Box\varphi_1, ..., \Box\varphi_k, \Theta$ where $\Theta \subseteq Lit$ is consistent and $k \geq 1$. The rule applied at $\mathbb{P}(n)$ is $(\Box)$ which generates a node u labelled by Δ, φ_i for $1 \leq i \leq k$. By assumption $\mathbb{P}'(n)$ is labelled by $\overline{D(\Diamond\Delta)}, \overline{D(\Box\varphi_1)}, ..., \overline{D(\Box\varphi_k)}, \overline{D(\Theta)}$ which is

$$\Box\overline{D(\Delta)}, \Diamond\overline{D(\varphi_1)}, ..., \Diamond\overline{D(\varphi_k)}, \overline{D(\Theta)}$$

Observe that $\overline{D(\Theta)} \subseteq Lit$ is consistent. So the rule applied at $\mathbb{P}'(n)$ is (mod) which generates k nodes $u_1', ..., u_k'$ labelled by $\overline{D(\Delta)}, \overline{D(\varphi_j)}$ for $1 \leq j \leq k$ respectively. Let $\mathbb{P}(n+1) = u$ and let $\mathbb{P}'(n+1) = u_i'$. By construction $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1)$ and $\mathbb{P}'(0)...\mathbb{P}'(n)\mathbb{P}'(n+1)$ are corresponding.

– Suppose $\mathbb{P}(n)$ is labelled by Δ where the least formula in Δ according to e to which a rule can be applied is a fixed-point variable Z identifying the formula $\varphi(Z)$. The rule applied at $\mathbb{P}(n)$ is therefore (Z) which generates a node u labelled by $\Delta', \varphi(Z)$ where $\Delta' = \Delta - \{Z\}$. By assumption $\mathbb{P}'(n)$ is labelled by $\overline{D(\Delta)}$ and the least formula in $\overline{D(\Delta)}$ according to $\hat{e}$ to which a rule can be applied is Z where Z identifies $\overline{D(\varphi(Z))}$. So the rule applied at $\mathbb{P}'(n)$ is (Z) which generates a node u' labelled by $\overline{D(\Delta')}, \overline{D(\varphi)}$. Let $\mathbb{P}(n+1) = u$ and $\mathbb{P}'(n+1) = u'$. By construction $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1)$ and $\mathbb{P}'(0)...\mathbb{P}'(n)\mathbb{P}'(n+1)$ are corresponding.

– Suppose $\mathbb{P}(n)$ is labelled by Δ where the least formula in Δ according to e to which a rule can be applied is $\mu Z.\varphi(Z)$ or $\nu Z.\varphi(Z)$. The rule applied at $\mathbb{P}(n)$ is thus (μ) or (ν) which generates a new node u labelled by Δ', Z where $\Delta' = \Delta - \{\mu Z.\varphi(Z)\}$ (or $\nu Z.\varphi(Z)$). By assumption $\mathbb{P}'(n)$ is labelled by $\overline{D(\Delta)}$ and the least formula in $\overline{D(\Delta)}$ according to $\hat{e}$ to which a rule can be applied is $\overline{D(\mu Z.\varphi(Z))} = \nu Z.\overline{D(\varphi(\neg Z))}$ or $\overline{D(\nu Z.\varphi(Z))} = \mu Z.\overline{D(\varphi(\neg Z))}$. Thus the rule applied at $\mathbb{P}'(n)$ is (ν) or (μ) respectively which generates the node u' labelled by $\overline{D(\Delta')}, Z$. Let $\mathbb{P}(n+1) = u$ and $\mathbb{P}'(n+1) = u'$. By construction $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1)$ and $\mathbb{P}'(0)...\mathbb{P}'(n)\mathbb{P}'(n+1)$ are corresponding.

Therefore the path $\mathbb{P}$ corresponds to the path $\mathbb{P}'$. Since t was an arbitrary pre-proof for Γ (in normal form) we conclude that every pre-proof for Γ in normal form with respect to e has a path that corresponds to some path through t'. Recall that t' is a tableau for $D(\Gamma)$. Thus by lemma 5.3.7 every path - and in particular $\mathbb{P}'$ - through t' is satisfying. So by lemma 5.3.11 the path $\mathbb{P}$ is invalid. Hence every pre-proof for Γ in normal form with respect to e has an invalid path, which implies that Γ does not have a proof in normal form with respect to e by lemma 5.3.4. □

Corollary 5.3.23. If $D(\Gamma)$ has a tableau, then Γ does not have a proof.

Proof. Suppose $D(\Gamma)$ has a tableau. Let e be an arbitrary enumeration for $Sub(\Gamma)$. Then $\hat{e}$ is an enumeration for $D(\Gamma)$. By lemma 5.3.19 $D(\Gamma)$ has a tableau in normal form with respect to $\hat{e}$. So by the previous theorem Γ does not have a proof in normal form with respect to e. The contraposition of lemma 5.3.21 thus implies that Γ does not have a proof. □

Theorem 5.3.24 (Soundness of **DT**)**.** If Γ is a sequent and $\mathbf{DT} \vdash \Gamma$, then Γ is valid.

Proof. We proceed by contraposition. Suppose Γ is not valid. Thus $D(\Gamma)$ is satisfiable. The completeness theorem for **T** implies that $D(\Gamma)$ has a tableau. By the previous corollary we conclude that Γ does not have a proof and so $\mathbf{DT} \nvdash \Gamma$. $\square$

5.4 The sequent calculus DT′

In this section we introduce the sequent calculus **DT′** which is a slight variation of **DT**. Recall the modality rule of **DT**:

$$\frac{\Gamma, \varphi_i}{\Diamond\Gamma, \Box\varphi_1, ..., \Box\varphi_n, \Theta}\ (\Box)$$

This rule essentially combines two rules into one:

1. A modality rule which given as premises a sequent Γ (possibly empty) and a formula φ introduces box-distribution: Formulas in Γ are bound by a diamond operator and φ is bound by a box operator.

$$\frac{\Gamma, \varphi}{\Diamond\Gamma, \Box\varphi}$$

2. A weakening rule which given an arbitrary sequent Γ introduces an arbitrary finite number of boxed formulas as well as an arbitrary finite set of consistent literals:

$$\frac{\Gamma}{\Gamma, \Box\psi_1, ..., \Box\psi_n, \Theta}$$

While it is quite standard to combine modality rules with weakening rules, our version of weakening is rather unusual. It is much more common to allow weakening with arbitrary finite sets of formulas instead of weakening only with formulas of a very specific form. The choice of this version of the $(\Box)$-rule in **DT** was motivated by its close connection to the (mod)-rule of the tableaux system **T**. In this section, we replace the rule $(\Box)$ by a more standard rule combining modality distribution with weakening by arbitrary side sequents. Let $(\Box')$ be the following rule:

$$\frac{\Gamma, \varphi}{\Diamond\Gamma, \Box\varphi, \Sigma}\ (\Box')$$

where Σ is a finite set of (arbitrary) L_μ-formulas. Clearly this rule is stronger than our previous rule $(\Box)$. Indeed $(\Box)$ is a special case of $(\Box')$ where Σ consists of boxed formulas and consistent literals only. Replacing $(\Box)$ in **DT** by $(\Box')$ results in the sequent calculus **DT′**.

Table 5.2: The sequent calculus $\mathbf{DT}'$

$$\frac{}{\Gamma, P, \neg P}\ (\mathrm{A}) \qquad \frac{\Gamma, \varphi_0 \quad \Gamma, \varphi_1}{\Gamma, \varphi_0 \wedge \varphi_1}\ (\wedge) \qquad \frac{\Gamma, \varphi_0, \varphi_1}{\Gamma, \varphi_0 \vee \varphi_1}\ (\vee)$$

$$\frac{\Gamma, Z}{\Gamma, \mu Z.\varphi(Z)}\ (\mu) \qquad \frac{\Gamma, Z}{\Gamma, \nu Z.\varphi(Z)}\ (\nu) \qquad \frac{\Gamma, \varphi(Z)}{\Gamma, Z}\ (Z)$$

$$\frac{\Gamma, \varphi}{\Diamond\Gamma, \Box\varphi, \Sigma}\ (\Box')$$

Definition 5.4.1. The sequent calculus $\mathbf{DT}'$ consists of the following axioms and rules:

1. The axiom (A) and the rules
2. $(\wedge), (\vee), (\Box'), (\mu), (\nu)$ and (Z)

and is depicted in table 5.2.

$\mathbf{DT}'$-*pre-proofs*, *paths*, *traces* and $\mathbf{DT}'$-*proofs* are defined as for $\mathbf{DT}$, see section 5.2. We write $\mathbf{DT}' \vdash \Gamma$ if and only if Γ has a $\mathbf{DT}'$-proof.

5.5 Soundness and completeness of $\mathbf{DT}'$

This section establishes the soundness and completeness of $\mathbf{DT}'$ by using the soundness and completeness results established for $\mathbf{DT}$. As mentioned above, the rule $(\Box)$ is a special case of the rule $(\Box')$. Therefore every $\mathbf{DT}$-derivation is also a $\mathbf{DT}'$-derivation. This directly yields that $\mathbf{DT}'$ is complete.

Theorem 5.5.1 (Completeness of $\mathbf{DT}'$)**.** If Γ is valid, then $\mathbf{DT}' \vdash \Gamma$.

Soundness is established by showing that every sequent which is $\mathbf{DT}'$-provable is also provable in $\mathbf{DT}$. To do so we use once again a suitable notion of corresponding paths in $\mathbf{DT}'$- and $\mathbf{DT}$-proofs. The only tricky case occurs when the $(\Box')$-rule is applied. The first problem is that the rule can be applied when a single box-formula is present at the sequent while for the rule $(\Box)$ every formula in the sequent has to be decomposed first until only formulas in the scope of a modal operator and consistent literals remain. Thus when we have a node in a $\mathbf{DT}'$-proof where the box-rule is applied, we first have to decompose every formula in the corresponding node in the $\mathbf{DT}$-proof before we can apply the box-rule there. The second and bigger problem is that we can eliminate diamond-formulas by applying $(\Box')$ while this is not possible when applying $(\Box)$. Suppose some node where $(\Box')$ is applied is labelled by $\Diamond\Gamma, \Box\varphi, \Sigma$ where Σ contains some formula $\Diamond\psi$. Thus in the corresponding proof, after decomposing every formula in Σ and then applying ($\Box$) we are left with a sequent of the form Γ, φ, ψ instead of just Γ, φ. Finally, we also have to take into account that it is possible to apply the rule ($\Box'$) even when the current node is labelled by inconsistent literals, while in the corresponding **DT**-proof the corresponding path has to end in that node. We start by introducing a refined notion of corresponding paths.

Definition 5.5.2. Suppose $t = (V, \rightarrow, \lambda)$ is a **DT**$'$-pre-proof for Γ and $t' = (V', \rightarrow', \lambda')$ is a **DT**-pre-proof for Γ.

- ▷ Let $u \in V$ and $u' \in V'$. The nodes u and u' are *corresponding* if $\lambda(u) \subseteq \lambda'(u')$.
- ▷ Let $\mathbb{P}$ be a path through t and $\mathbb{P}'$ a path through t'. The paths $\mathbb{P}$ and $\mathbb{P}'$ are *corresponding*, if all of the following conditions hold:
  1. Either both $\mathbb{P}$ and $\mathbb{P}'$ are finite or both $\mathbb{P}$ and $\mathbb{P}'$ are infinite.
  2. For each $i \in \omega$ there exists $j \geq i$ such that $\mathbb{P}(i)$ corresponds to $\mathbb{P}'(j)$ (assuming that $\mathbb{P}(i)$ exists).
  3. If $\mathbb{P}(i)$ corresponds to $\mathbb{P}'(j)$ and $\mathbb{P}(n)$ corresponds to $\mathbb{P}'(m)$ where $i < n$, then $j < m$.
  4. If $\mathbb{P}$ and $\mathbb{P}'$ are finite and $\mathbb{P}$ ends in $\mathbb{P}(n)$ which corresponds to $\mathbb{P}'(m)$, then $\mathbb{P}'$ ends in $\mathbb{P}'(m)$.

Observe that if a path $\mathbb{P}$ in t corresponds to a path $\mathbb{P}'$ in t', then we require that each node in $\mathbb{P}$ corresponds to some node in $\mathbb{P}'$. However, we do not require that the converse holds as well. Thus it is possible that some nodes in $\mathbb{P}'$ do not correspond to any nodes in $\mathbb{P}$. Moreover, point 3. ensures that the correspondence relation is monotone. Point 4. is strictly speaking not necessary for the proof, as we define the corresponding paths in such a way that point 4. is always fulfilled anyway. We add the restriction to the definition to prevent confusion.

Theorem 5.5.3. Let Γ be a sequent. If **DT**$' \vdash \Gamma$, then **DT** $\vdash \Gamma$.

Proof. Suppose $t = (V, \rightarrow, \lambda)$ is a **DT**$'$-proof of Γ. We show how to construct a **DT**-proof $t' = (V', \rightarrow', \lambda')$ of Γ.

- ▷ The root $r_{t'}$ of t' is labelled by Γ. Notice that r_t corresponds to $r_{t'}$.
- ▷ Suppose we have constructed $\mathbb{P}'(0)...\mathbb{P}'(j)$ where $\mathbb{P}'(0) = r_{t'}$ which corresponds to the initial segment $\mathbb{P}(0)...\mathbb{P}(i)$ of some path $\mathbb{P}$ through t where $i \leq j$. Moreover we assume that $\mathbb{P}'(j)$ corresponds to $\mathbb{P}(i)$. We distinguish two cases:
  - Suppose the rule applied at $\mathbb{P}(i)$ is $(*)$ for $* \in \{\wedge, \vee, Z, \mu, \nu\}$. Then we apply the same rule to the same formula labelling $\mathbb{P}'(j)$ to generate the node $\mathbb{P}'(j+1)$ which corresponds to $\mathbb{P}(i+1)$. Notice that $\mathbb{P}'(0)...\mathbb{P}'(j)\mathbb{P}'(j+1)$ corresponds to $\mathbb{P}(0)...\mathbb{P}(i)\mathbb{P}(i+1)$.

– Suppose $\mathbb{P}(i)$ is labelled by $\Diamond\Pi, \Box\varphi_1, ..., \Box\varphi_k, \Sigma$ where $k \geq 1$ and Σ is an arbitrary sequent and the rule applied is $(\Box')$ which generates the node $\mathbb{P}(i+1)$ labelled by Π, φ_d for $d \leq k$:

$$\frac{\Pi, \varphi_d}{\Diamond\Pi, \Box\varphi_1, ..., \Box\varphi_k, \Sigma}\ (\Box')$$

By assumption $\mathbb{P}(i)$ corresponds to $\mathbb{P}'(j)$, which implies that $\mathbb{P}'(j)$ is labelled by $\Diamond\Pi, \Box\varphi_1, ..., \Box\varphi_k, \Sigma, \Omega$ where Ω is some side-sequent. We first apply rules to decompose each formula in Σ, Ω until we are left with boxed formulas, diamond formulas and literals only. Notice that decomposing these formulas might split up the branch in case we have to apply the rule $(\wedge)$. Thus after decomposing these formulas we have constructed nodes $u_1, ..., u_n$ where u_l for $1 \leq l \leq n$ is labelled by

$$\Diamond\Pi, \Box\varphi_1, ..., \Box\varphi_k, \Diamond\Delta_l, \Theta_l$$

where $\Delta_l \subseteq \{\psi | \Diamond\psi \in Sub(\Sigma) \cup Sub(\Omega)\}$ and $\Theta_l \subseteq Lit(\Sigma) \cup Lit(\Omega)$ where $Lit(\Sigma)$ is the set of all literals that occur as subformulas in Σ. Suppose that there are s_l steps between $\mathbb{P}'(j)$ and u_l. Suppose without loss of generality that we have extended $\mathbb{P}'$ for s_l-steps such that $\mathbb{P}'(j+s_l) = u_l$ (the case where the path goes through another node $u_{l'}$ is identical). Then we distinguish two cases:

1. Suppose the literals in Θ_l are inconsistent. Then the path $\mathbb{P}'$ ends in u_l and we call $\mathbb{P}'$ a non-corresponding path.
2. Suppose the literals in Θ_l are consistent. Then apply the rule $(\Box)$ to generate the node $\mathbb{P}'(j+s_l+1)$ which is labelled by Π, φ_d, Δ_l. Notice that $\mathbb{P}(i+1)$ corresponds to $\mathbb{P}'(j+s_l+1)$ and moreover the initial segment $\mathbb{P}(0)...\mathbb{P}(i)\mathbb{P}(i+1)$ corresponds to $\mathbb{P}'(0)...\mathbb{P}'(j)...\mathbb{P}'(j+s_l)\mathbb{P}'(j+s_l+1)$.

Therefore we have shown how to construct the tree t'. First of all, notice that t' is generated by the rules of **DT** and is therefore a finite branching tree. The root of t' is labelled by Γ. Furthermore, every path through t' is either non-corresponding or corresponds to a path through t (observe that several paths in t' might correspond to the same path in t). We now show that every path $\mathbb{P}'$ through t' is valid:

1. Suppose $\mathbb{P}'$ is non-corresponding. Thus $\mathbb{P}'$ is a finite path that ends in a leaf which is labelled by inconsistent literals. This implies that the leaf is axiomatic. Therefore $\mathbb{P}'$ is valid.

2. Suppose $\mathbb{P}'$ corresponds to a finite path $\mathbb{P}$. Then $\mathbb{P}'$ is finite itself and the leaf $\mathbb{P}(i)$ of $\mathbb{P}$ corresponds to the leaf $\mathbb{P}'(j)$ of $\mathbb{P}'$. Therefore $\lambda(\mathbb{P}(i)) \subseteq \lambda'(\mathbb{P}'(j))$. Since t is a proof we have that $\mathbb{P}(i)$ is labelled by inconsistent literals and thus so is $\mathbb{P}'(j)$, which implies that $\mathbb{P}'(j)$ is an axiomatic leaf. Therefore $\mathbb{P}'$ is valid.

3. Suppose $\mathbb{P}'$ corresponds to an infinite path $\mathbb{P}$. Thus $\mathbb{P}'$ is an infinite path itself. Since t is a proof there exists an infinite ν-trace through $\mathbb{P}$ and since $\mathbb{P}$ corresponds to $\mathbb{P}'$, for each i there exists $j \geq i$ such that $\lambda(\mathbb{P}(i)) \subseteq \lambda'(\mathbb{P}'(j))$. Therefore there exists an infinite ν-trace through $\mathbb{P}'$ as well, which implies that $\mathbb{P}'$ is valid.

All together we conclude that t' is a **DT**-proof of Γ. □

Theorem 5.5.4 (Soundness of $\mathbf{DT}'$)**.** If $\mathbf{DT}' \vdash \Gamma$, then Γ is valid.

Proof. Suppose Γ is a sequent and $\mathbf{DT}' \vdash \Gamma$. Theorem 5.5.3 implies that $\mathbf{DT} \vdash \Gamma$. The soundness theorem 5.3.24 for **DT** then implies that Γ is valid. □

5.6 The two-sided sequent calculus 2DT

The last part of this chapter is devoted to introduce the system **2DT**, a sequent calculus that operates on two-sided sequents instead of the one-sided sequents considered so far.

Definition 5.6.1. A *two-sided sequent* is an ordered pair $\Gamma \Rightarrow \Delta$ where Γ, Δ are finite sets of L_μ-formulas.

The switch from one-sided sequents to two-sided sequents is motivated by the goal to establish Craig-interpolation for $\Sigma_1^\mu \cup \Pi_1^\mu$ in the next chapter, where two-sided sequents are a natural framework to represent interpolation. For the rest of this chapter we refer to two-sided sequents just as sequents. In a sequent $\Gamma \Rightarrow \Delta$ it is allowed for Γ or Δ to be empty. The interpretation of a two-sided sequent is given in terms of classical implication $\rightarrow$ defined by

$$\varphi \rightarrow \psi := D(\varphi) \vee \psi$$

Definition 5.6.2. The *interpretation* $\mathcal{I}$ of a sequent $\Gamma \Rightarrow \Delta$ is given by:

$$\mathcal{I}(\Gamma \Rightarrow \Delta) := \bigwedge \Gamma \rightarrow \bigvee \Delta$$

where $\bigwedge \emptyset := \top$ and $\bigvee \emptyset := \bot$. We call a sequent $\Gamma \Rightarrow \Delta$ *valid* if and only if $\mathcal{I}(\Gamma \Rightarrow \Delta)$ is valid.

Observe that the completely empty sequent $\Rightarrow$ is unsatisfiable, as $\mathcal{I}(\Rightarrow) = \top \rightarrow \bot \equiv \bot$. By definition of the translation D the following holds:

$$\begin{aligned} \Gamma \Rightarrow \Delta \text{ is valid } &\Leftrightarrow \mathcal{I}(\Gamma \Rightarrow \Delta) \text{ is valid} \\ &\Leftrightarrow \bigwedge \Gamma \rightarrow \bigvee \Delta \text{ is valid} \\ &\Leftrightarrow \bigvee D(\Gamma) \vee \bigvee \Delta \text{ is valid} \end{aligned}$$

The sequent calculus **2DT** consists of left-side and right-side rules for each operator and connective. Furthermore, it consists of a total of four axioms, where we consider a sequent to be an axiom if either $P, \neg P$ occurs on the left or the right side for arbitrary $P \in Prop$ or a literal occurs on both sides. It follows directly from the definition of the interpretation of a sequent that all four axioms are valid. If $P, \neg P$ occurs on the right, then the right side is equivalent to $\top$ as it is interpreted as a disjunction. Since $\varphi \to \top$ is valid for any φ, the axiom is valid. If $P, \neg P$ occurs on the left side, then the left side is equivalent to $\bot$ as it is interpreted as a conjunction and $\bot \to \varphi$ is once again valid for any formula φ. Finally, if a literal occurs on both sides, then since $P \to P$ and $\neg P \to \neg P$ are both valid, so are the axioms. Apart from adding more rules and axioms to deal with formulas on both sides of the sequent arrow, we also formulate new requirements for infinite paths to be valid. The basic slogan is that an infinite ν-trace occurring on the right side is sufficient for a path to be valid and so is an infinite μ-trace on the left.

Table 5.3: The sequent calculus **2DT**

$$\frac{}{\Gamma, P \Rightarrow P, \Delta}\ (\mathrm{Ax})_1 \qquad \frac{}{\Gamma \Rightarrow P, \neg P, \Delta}\ (\mathrm{Ax})_2$$

$$\frac{}{\Gamma, \neg P \Rightarrow \neg P, \Delta}\ (\mathrm{Ax})_3 \qquad \frac{}{\Gamma, P, \neg P \Rightarrow \Delta}\ (\mathrm{Ax})_4$$

$$\frac{\Gamma, \varphi_0, \varphi_1 \Rightarrow \Delta}{\Gamma, \varphi_0 \wedge \varphi_1 \Rightarrow \Delta}\ (\wedge)_L \qquad \frac{\Gamma \Rightarrow \varphi_0, \Delta \quad \Gamma \Rightarrow \varphi_1, \Delta}{\Gamma \Rightarrow \varphi_0 \wedge \varphi_1, \Delta}\ (\wedge)_R$$

$$\frac{\Gamma, \varphi_0 \Rightarrow \Delta \quad \Gamma, \varphi_1 \Rightarrow \Delta}{\Gamma, \varphi_0 \vee \varphi_1 \Rightarrow \Delta}\ (\vee)_L \qquad \frac{\Gamma \Rightarrow \varphi_0, \varphi_1, \Delta}{\Gamma \Rightarrow \varphi_0 \vee \varphi_1, \Delta}\ (\vee)_R$$

$$\frac{\Gamma, \varphi \Rightarrow \Delta}{\Theta, \Box\Gamma, \Diamond\varphi \Rightarrow \Diamond\Delta, \Sigma}\ (\Box)_L \qquad \frac{\Gamma \Rightarrow \varphi, \Delta}{\Theta, \Box\Gamma \Rightarrow \Box\varphi, \Diamond\Delta, \Sigma}\ (\Box)_R$$

$$\frac{\Gamma, Z \Rightarrow \Delta}{\Gamma, \mu Z.\varphi(Z) \Rightarrow \Delta}\ (\mu)_L \qquad \frac{\Gamma \Rightarrow Z, \Delta}{\Gamma \Rightarrow \mu Z.\varphi(Z), \Delta}\ (\mu)_R$$

$$\frac{\Gamma, Z \Rightarrow \Delta}{\Gamma, \nu Z.\varphi(Z) \Rightarrow \Delta}\ (\nu)_L \qquad \frac{\Gamma \Rightarrow Z, \Delta}{\Gamma \Rightarrow \nu Z.\varphi(Z), \Delta}\ (\nu)_R$$

$$\frac{\Gamma, \varphi(Z) \Rightarrow \Delta}{\Gamma, Z \Rightarrow \Delta}\ (Z)_L \qquad \frac{\Gamma \Rightarrow \varphi(Z), \Delta}{\Gamma \Rightarrow Z, \Delta}\ (Z)_R$$

Definition 5.6.3. The sequent calculus **2DT** consists of the following axioms and rules:

1. The axioms $(Ax)_1$, $(Ax)_2$, $(Ax)_3$ and $(Ax)_4$

2. Left and right side rules for the Boolean connectives $\wedge$ and $\vee$
3. Left and right side modality rules
4. Left and right side rules for fixed-point operators and fixed point variables

and is depicted in table 5.3.

As usual, every side-sequent occurring in a rule is allowed to be empty. Moreover, in the rule $(\Box)_L$ only the formula $\Diamond\varphi$ on the left-side is required to apply the rule. Similarly, in the rule $(\Box)_R$ the only formula required is $\Box\varphi$ occurring on the right. The side-sequents Θ and Σ are arbitrary finite sets of L_μ-formulas in both rules. As before, every variable occurring in a sequent is assumed to identify a unique formula.

Definition 5.6.4. A *pre-proof* for $\Gamma \Rightarrow \Delta$ is a (possibly infinite) tree $t = (V, \to, \lambda)$ whose root is labelled by $\Gamma \Rightarrow \Delta$ and which is built according to the rules depicted in table 5.3, such that every leaf is labelled by an axiom or by a sequent of the form $\Theta, \Box\Gamma \Rightarrow \Diamond\Delta, \Sigma$, where $\Theta \subseteq Lit$ and $\Sigma \subseteq Lit$.

The requirement for leafs ensures that pre-proofs are maximal. Every finite path either ends in an axiom or in a leaf where no more formula can be applied. *Paths* and *traces* are defined as for **DT**; we refer to section 5.2. Notice that **2DT** contains no rules that push formulas from one side of the arrow to the other. That implies that a trace starting on one side of the sequent arrow stays on that side. This simplifies the requirements that are put on infinite paths in order to be valid, as we do not have to deal with infinite traces passing both through the left and the right side. Observe that the $(\wedge)_R$-rule is dual to the $(\vee)_L$-rule in the sense that they both generate branching of degree 2. Similarly, the $(\vee)_R$-rule is dual to $(\wedge)_L$. The modality rules introduce weakening on both sides as well as box distribution. Notice that it is allowed to weaken with arbitrary side-sequents. The fixed point rules are straightforward generalizations of the fixed point rules in $\mathbf{DT}'$ on both sides.

Definition 5.6.5. A *proof* for $\Gamma \Rightarrow \Delta$ is a pre-proof $t = (V, \to, \lambda)$ for $\Gamma \Rightarrow \Delta$ for which the following holds:

1. Every leaf of t is labelled by an axiom.
2. Every infinite path through t contains an infinite μ-trace on the left or an infinite ν-trace on the right.

We write $\mathbf{2DT} \vdash \Gamma \Rightarrow \Delta$ if and only if there exists a **2DT**-proof for $\Gamma \Rightarrow \Delta$. For a L_μ-formula φ, we say that φ is provable if and only if $\mathbf{2DT} \vdash \Rightarrow \varphi$. Similarly, a one-sided sequent Γ is said to be provable if and only if $\mathbf{2DT} \vdash \Rightarrow \Gamma$. As for tableaux, we introduce the notion of principal and residual formulas. Each **2DT**-rule which is not a modality rule works on a single formula in the lower sequent which is decomposed or regenerated. We call these relevant formulas in the premise and the conclusion the *distinguished formulas* of the rule. For every rule apart from the modality rules, the distinguished formula in the lower sequent is called *principal* and the distinguished formula(s) in the upper sequent is (are) called *residual*. In case of the rules $(\Box)_L$ and $(\Box)_R$, only the formulas in the lower sequent which survive the rule are *principal* (that is, the formulas in $\Box\Gamma, \Diamond\Delta$ and $\bullet\varphi$ for $\bullet \in \{\Box, \Diamond\}$) and every formula in the upper sequent is *residual*. For example in the rule

$$\frac{\Gamma \Rightarrow \varphi_0, \Delta \qquad \Gamma \Rightarrow \varphi_1, \Delta}{\Gamma \Rightarrow \varphi_0 \wedge \varphi_1, \Delta} (\wedge)_R$$

the principal formula is $\varphi_0 \wedge \varphi_1$ and the residual formulas are φ_0 and φ_1 in the upper sequents.

5.7 Soundness and completeness of 2DT

In order to prove soundness and completeness of **2DT**, we use the close connection between **2DT** and **DT′** as well as the already established result that **DT′** is sound and complete. The correspondence between **2DT** and **DT′** is given by the following theorem:

Theorem 5.7.1. A one-sided sequent Γ is **2DT**-provable if and only if Γ is **DT′**-provable.

Proof. Recall that there exists no rule in **2DT** which pushes a formula from one side of the sequent arrow to the other and that a one-sided sequent Γ is **2DT**-provable means that **2DT** $\vdash \Rightarrow \Gamma$. If there is a proof of $\Rightarrow \Gamma$, the only rules applied in this proof are therefore rules for the right side of the sequent arrow and $(Ax)_2$. Moreover, throughout the proof, the left-side is always empty. Observe that if the left-side of a sequent is empty, then every right-side rule in **2DT** coincides with the same rule in **DT′**; that is, $(\wedge)_R$ coincides with $(\wedge)$, $(\vee)_R$ coincides with $(\vee)$, $(\Box)_R$ coincides with $(\Box')$ and so on. Thus, given a **2DT**-proof of $\Rightarrow \Gamma$, one obtains a **DT′**-proof of Γ by simply removing every sequent arrow in the tree. For the other direction, given a **DT′**-proof of Γ, one obtains a **2DT**-proof for $\Rightarrow \Gamma$ by simply adding a sequent arrow to the left of each sequent labelling a node in the proof tree. □

Observe that theorem 5.7.1, despite establishing a useful connection between **2DT** and **DT′**, is not quite enough to prove soundness and completeness. Using theorem 5.7.1 and the soundness and completeness of **DT′**, we can deduce that $\Rightarrow \Gamma$ is **2DT**-provable if and only if $\Rightarrow \Gamma$ is valid. Our goal, however, is to prove that $\Gamma \Rightarrow \Delta$ is **2DT**-provable if and only if $\Gamma \Rightarrow \Delta$ is valid. What we require is a bridge theorem that establishes that a sequent $\Gamma \Rightarrow \Delta$ is provable if and only if $\Rightarrow D(\Gamma), \Delta$ is provable. We start by proving one direction.

Remark 5.7.2. Recall that $\overline{\varphi}$ denotes the formula φ, where every negated formula $\neg Z$ in φ is replaced by Z. For the remainder of this section we denote the set $\overline{D(\Gamma)}$ simply by $D(\Gamma)$ in order to obtain a more concise presentation.

Theorem 5.7.3. Let $\Gamma \Rightarrow \Delta$ be a sequent. If **2DT** $\vdash \Rightarrow D(\Gamma), \Delta$, then **2DT** $\vdash \Gamma \Rightarrow \Delta$.

The main difficulty in proving the theorem is that there might be some formula $\varphi \in \Gamma$ such that $D(\varphi) \in \Delta$. In that case there is some information lost in the sequent $\Rightarrow D(\Gamma), \Delta$ compared to $\Gamma \Rightarrow \Delta$, as the formula φ occurs in $\Gamma \Rightarrow \Delta$ on the left and its dual $D(\varphi)$ on the right, while in $\Rightarrow D(\Gamma), \Delta$ the formula $D(\varphi)$ only occurs once. For solving this problem we introduce a new notion of corresponding paths in **2DT**-proofs.

Definition 5.7.4. Let $t = (V, \to, \lambda)$ be a **2DT**-pre-proof for $\Gamma \Rightarrow \Delta$ and $t' = (V', \to', \lambda')$ a **2DT**-pre-proof for $\Rightarrow D(\Gamma), \Delta$. Let $u \in V$ and $u' \in V'$. We say that u and u' are *corresponding* - written $u \leftrightsquigarrow u'$ - if and only if

$$\lambda(u) = (\Gamma', \Pi \Rightarrow \Delta') \text{ and } \lambda'(u') = (\Rightarrow D(\Gamma'), \Delta')$$

where Π is an arbitrary finite set of formulas. We call two paths $\mathbb{P}$ through t and $\mathbb{P}'$ through t' *corresponding* - written $\mathbb{P} \leftrightsquigarrow \mathbb{P}'$ - if and only if

1. both $\mathbb{P}$ and $\mathbb{P}'$ are finite, have the same length and $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(n)$ for all $n \leq lth(\mathbb{P})$ or
2. both $\mathbb{P}$ and $\mathbb{P}'$ are infinite and $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(n)$ for all $n \in \omega$.

The general proof strategy is based on the slogan that if a formula φ occurs on the left and its dual $D(\varphi)$ on the right, then only information from one of these two formulas is needed. We decide to use in such cases only information from the right, ignoring the formula on the left. That is, we show how to construct a proof t for $\Gamma \Rightarrow \Delta$ from the proof t' for $\Rightarrow D(\Gamma), \Delta$, where whenever a rule in t' is applied to a formula $D(\varphi)$, such that on the corresponding node in t the formula φ occurs on the left and $D(\varphi)$ on the right, then we only apply the corresponding rule on the right, ignoring the formula φ on the left. Thus, in the next node we consider φ to belong to Π. We show that this procedure still leads to a proof for $\Gamma \Rightarrow \Delta$, despite ignoring information on the left. From now on when given a sequent $\Gamma \Rightarrow \Delta$, we write this sequent as $\Gamma_1, \Sigma \Rightarrow D(\Sigma), \Delta_1$ where we assume that $D(\Gamma_1) \cap \Delta_1 = \emptyset$. If there is no formula $\varphi \in \Gamma$ such that $D(\varphi) \in \Delta$, then the sets Σ and $D(\Sigma)$ are empty.

Proof. (of theorem 5.7.3) Let $\Gamma, \Sigma \Rightarrow D(\Sigma), \Delta$ be a sequent where $D(\Gamma) \cap \Delta = \emptyset$. Suppose $t' = (V', \to', \lambda')$ is a proof for $\Rightarrow D(\Gamma), D(\Sigma), \Delta$. We construct a proof $t = (V, \to, \lambda)$ for $\Gamma, \Sigma \Rightarrow D(\Sigma), \Delta$ as follows:

- ▷ The root r_t of t is labelled by $\Gamma, \Sigma \Rightarrow D(\Sigma), \Delta$. Notice that $r_t \leftrightsquigarrow r_{t'}$ where $r_{t'}$ is the root of t'.
- ▷ Suppose we have constructed $\mathbb{P}(0)\mathbb{P}(1)...\mathbb{P}(n)$ which corresponds to the initial segment of a path $\mathbb{P}'(0)\mathbb{P}'(1)...\mathbb{P}'(n)$ through t'. We show how to extend the path:
  - Suppose $\mathbb{P}'(n)$ is labelled by $\Rightarrow D(\Gamma'), D(\Sigma'), \Delta', \varphi \vee \psi$. The rule applied at $\mathbb{P}'(n)$ is $(\vee)_R$ and $\mathbb{P}'(n+1)$ is labelled by $\Rightarrow D(\Gamma'), D(\Sigma'), \Delta', \varphi, \psi$. We distinguish three cases:

1. Suppose $\varphi \vee \psi \in D(\Gamma')$. Since $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(n)$, the node $\mathbb{P}(n)$ is labelled by $\Gamma', D(\varphi \vee \psi), \Sigma', \Pi \Rightarrow D(\Sigma'), \Delta'$ where $D(\Gamma') \cap \Delta' = \emptyset$ and Π is some set of side-formulas. Notice that $D(\varphi \vee \psi) = D(\varphi) \wedge D(\psi)$. Apply the rule $(\wedge)_L$ to generate a new node u labelled by $\Gamma', D(\varphi), D(\psi), \Sigma', \Pi \Rightarrow D(\Sigma'), \Delta'$ and let $\mathbb{P}(n+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(n+1)$.
2. Suppose $\varphi \vee \psi \in \Delta'$. Then $\mathbb{P}(n)$ is labelled by $\Gamma', \Sigma', \Pi \Rightarrow D(\Sigma'), \Delta', \varphi \vee \psi$. Apply the rule $(\vee)_R$ to generate the node u labelled by $\Gamma', \Sigma', \Pi \Rightarrow D(\Sigma'), \Delta', \varphi, \psi$ and let $\mathbb{P}(n+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(n+1)$.
3. Suppose $\varphi \vee \psi \in D(\Sigma')$. Then $\mathbb{P}(n)$ is labelled by

$$\Gamma', \Sigma', D(\varphi) \wedge D(\psi), \Pi \Rightarrow D(\Sigma'), \varphi \vee \psi, \Delta'$$

Apply the rule $(\vee)_R$ to generate the node u labelled by

$$\Gamma', \Sigma', D(\varphi) \wedge D(\psi), \Pi \Rightarrow D(\Sigma'), \varphi, \psi, \Delta'$$

and let $\mathbb{P}(n+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(n+1)$ as $D(\varphi) \wedge D(\psi)$ is now considered to belong to Π.

– Suppose $\mathbb{P}'(n)$ is labelled by $\Rightarrow D(\Gamma'), D(\Sigma'), \Delta', \varphi \wedge \psi$. The rule applied at $\mathbb{P}'(n)$ is $(\wedge)_R$ and $\mathbb{P}'(n+1)$ is without loss of generality labelled by $\Rightarrow D(\Gamma'), D(\Sigma'), \Delta', \varphi$. Again we distinguish three cases:

1. Suppose $\varphi \wedge \psi \in D(\Gamma')$. Then $\mathbb{P}(n)$ is labelled by

$$\Gamma', D(\varphi) \vee D(\psi), \Sigma', \Pi \Rightarrow D(\Sigma'), \Delta'$$

Apply the rule $(\vee)_L$ to generate two new nodes u and v labelled by $\Gamma', D(\varphi), \Sigma', \Pi \Rightarrow D(\Sigma'), \Delta'$ and $\Gamma', D(\psi), \Sigma', \Pi \Rightarrow D(\Sigma'), \Delta'$ respectively. Let $\mathbb{P}(n+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(n+1)$.
2. Suppose $\varphi \wedge \psi \in \Delta'$. Then $\mathbb{P}(n)$ is labelled by

$$\Gamma', \Sigma', \Pi \Rightarrow D(\Sigma'), \Delta', \varphi \wedge \psi$$

Apply the rule $(\wedge)_R$ to generate two new nodes u and v labelled by $\Gamma', \Sigma', \Pi \Rightarrow D(\Sigma'), \Delta', \varphi$ and $\Gamma', \Sigma', \Pi \Rightarrow D(\Sigma'), \Delta', \psi$ respectively. Let $\mathbb{P}(n+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(n+1)$.
3. Suppose $\varphi \wedge \psi \in D(\Sigma')$. Therefore $\mathbb{P}(n)$ is labelled by

$$\Gamma', \Sigma', D(\varphi) \vee D(\psi), \Pi \Rightarrow D(\Sigma'), \varphi \wedge \psi, \Delta'$$

Apply the rule $(\wedge)_R$ to generate two new nodes u and v which are labelled by $\Gamma', \Sigma', D(\varphi) \vee D(\psi), \Pi \Rightarrow D(\Sigma'), \varphi, \Delta'$ and $\Gamma', \Sigma', D(\varphi) \vee D(\psi), \Pi \Rightarrow D(\Sigma'), \psi, \Delta'$ respectively. Let $\mathbb{P}(n+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(n+1)$.

- The cases where $\mathbb{P}'(n)$ is labelled by a formula $\sigma Z.\varphi(Z)$ for $\sigma \in \{\mu, \nu\}$ or by Z where Z identifies $\varphi(Z)$ are similar to the first case; depending on whether the formulas are in $D(\Gamma'), D(\Sigma)$ or Δ' we apply the corresponding rule on the left or right and extend the path accordingly.
- Suppose $\mathbb{P}'(n)$ is labelled by $\Rightarrow D(\Gamma'), D(\Sigma'), \Delta', \Box\varphi$ where $D(\Gamma') = \Diamond D(\Gamma_1), D(\Gamma_2)$, $D(\Sigma') = \Diamond D(\Sigma_1), D(\Sigma_2)$ and $\Delta' = \Diamond\Delta_1, \Delta_2$. The rule applied is $(\Box)_R$ and $\mathbb{P}'(n+1)$ is labelled by $\Rightarrow D(\Gamma_1), D(\Sigma_1), \Delta_1, \varphi$. We distinguish three cases:
  1. Suppose $\Box\varphi \in D(\Gamma)$. This implies that $\Box\varphi \in D(\Gamma_2)$ and so $\mathbb{P}(n)$ is labelled by

     $$\Box\Gamma_1, \Gamma_2, \Diamond D(\varphi), \Box\Sigma_1, \Sigma_2, \Pi \Rightarrow \Diamond D(\Sigma_1), D(\Sigma_2), \Diamond\Delta_1, \Delta_2$$

     Apply the rule $(\Box)_L$ to generate the node u labelled by $\Gamma_1, D(\varphi), \Sigma_1 \Rightarrow D(\Sigma_1), \Delta_1$ and let $\mathbb{P}(n+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(n+1)$.
  2. Suppose $\Box\varphi \in \Delta$. This implies that $\Box\varphi \in \Delta_2$ and so that $\mathbb{P}(n)$ is labelled by

     $$\Box\Gamma_1, \Gamma_2, \Box\Sigma_1, \Sigma_2, \Pi \Rightarrow \Diamond D(\Sigma_1), D(\Sigma_2), \Diamond\Delta_1, \Delta_2, \Box\varphi$$

     Then apply the rule $(\Box)_R$ to generate the node u labelled by $\Gamma_1, \Sigma_1 \Rightarrow D(\Sigma)_1, \Delta_1, \varphi$ and let $\mathbb{P}(n+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(n+1)$.
  3. Suppose $\Box\varphi \in D(\Sigma)$. This implies that $\Box\varphi \in D(\Sigma_2)$ and therefore $\mathbb{P}(n)$ is labelled by

     $$\Box\Gamma_1, \Gamma_2, \Box\Sigma_1, \Sigma_2, \Diamond D(\varphi), \Pi \Rightarrow \Diamond D(\Sigma_1), D(\Sigma_2), \Box\varphi, \Diamond\Delta_1, \Delta_2$$

     Apply the rule $(\Box)_R$ to generate the node u labelled by $\Gamma_1, \Sigma_1 \Rightarrow D(\Sigma_1), \Delta_1, \varphi$ and let $\mathbb{P}(n+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(n+1)$.

We have shown how to construct the tree t from t' such that every path in t corresponds to a path in t'. Notice that the root of t is labelled by $\Gamma, \Sigma \Rightarrow D(\Sigma), \Delta$ and the tree is generated by applying the rules of **2DT**. Suppose u is a leaf of t and $\mathbb{P}(0)...\mathbb{P}(n) = u$ is the finite path leading to u. Let $\mathbb{P}'(0)...\mathbb{P}'(n)$ be its corresponding path. By definition of a corresponding path we have that $\mathbb{P}'(n)$ is a leaf of t'. Moreover it is an axiomatic leaf as t' is a proof. Hence $\mathbb{P}'(n)$ is labelled by $\Rightarrow D(\Gamma'), D(\Sigma'), \Delta', P, \neg P$. Since $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(n)$ this implies that $\mathbb{P}(n)$ is an axiomatic leaf as well, namely either $(Ax)_1, (Ax)_2, (Ax)_3$ or $(Ax)_4$ depending on whether P and $\neg P$ belong to $D(\Gamma')$ or to $D(\Sigma')$ or to Δ' or any combination of those. We conclude that every leaf of t is axiomatic, which implies further that t is a pre-proof. Next suppose $\mathbb{P}$ is an infinite path through t. Again there exists a corresponding path $\mathbb{P}'$ through t'. Since t' is a proof we have that $\mathbb{P}'$ contains an infinite ν-trace on the right. Since $\mathbb{P} \leftrightsquigarrow \mathbb{P}'$ this infinite ν-trace corresponds to an infinite ν-trace on the right or an infinite μ-trace on the left through $\mathbb{P}$. The only problematic case that might occur is when the infinite ν-trace on the right through $\mathbb{P}'$ starts in a $D(\Gamma)$-formula but switches to a $D(\Sigma)$-formula after a couple of steps. In this case we would start building the corresponding trace through $\mathbb{P}$ on the left but as soon as the trace switches to $D(\Sigma)$ we would continue the trace on the right. However, such a switch is only possible if the unravelled fixed point formula which originally belongs to $D(\Gamma)$ is a subformula of a formula in $D(\Sigma)$ or Δ. Otherwise, there is no possibility to introduce it on the right side. This implies that there is an infinite ν-trace on the right. Therefore every infinite branch of t has an infinite μ-trace on the left or an infinite ν-trace on the right. We conclude that t is a proof for $\Gamma, \Sigma \Rightarrow D(\Sigma), \Delta$. □

We have established that whenever there is a proof for $\Rightarrow D(\Gamma), \Delta$, there is also a proof for $\Gamma \Rightarrow \Delta$. For the converse direction we cannot proceed in the exact same way. Suppose we have a proof t for $\Gamma \Rightarrow \Delta$ and we want to construct a proof t' for $\Rightarrow D(\Gamma), \Delta$. The main problem is again that there might be some formula $\varphi \in \Gamma$ such that $D(\varphi) \in \Delta$. As we consider the proof t we must take care of situations where at some point in t there is rule applied to φ on the left and several steps later there is a rule applied to $D(\varphi)$ on the right or vice versa. As the formula $D(\varphi)$ occurs in $\Rightarrow D(\Gamma), \Delta$ only once, we have to decide whether we want to apply a rule to $D(\varphi)$ when the corresponding rule is applied to φ in t or when it is applied to $D(\varphi)$ in t. We decide to always apply a rule to $D(\varphi)$ in t' when the first time a rule is applied to one of φ or $D(\varphi)$ in t. The second time ignore the application of the rule and we do not not apply any rule. However, for that we need a way to keep track whether for some formula its dual formula on the other side has already been decomposed. Moreover, it can also happen that for some subformula of a formula there is a dual formula on the other side of the sequent arrow. Thus we also have to keep track of situations, where the subformula is freed and then has its dual on the other side. We will do this by introducing a marking on proofs that keeps track of these situations. Finally, we also have to slightly change the notion of a corresponding path, as it is now possible that when there is a step in a path through t, that nothing happens on its corresponding path through t'.

Definition 5.7.5. Let $t = (V, \to, \lambda)$ be a proof for $\Gamma \Rightarrow \Delta$. We define a marking on t as follows:

- ▷ The root r_t of t which is labelled by $\Gamma \Rightarrow \Delta$ has every formula in $\Gamma \cup \Delta$ unmarked.
- ▷ Suppose $\mathbb{P}(0)...\mathbb{P}(n)$ is an initial segment of a path $\mathbb{P}$ through t and we have marked $\mathbb{P}$ up to $\mathbb{P}(n)$. Moreover suppose $\lambda(\mathbb{P}(n)) = \Gamma_n \Rightarrow \Delta_n$ and the rule applied at $\mathbb{P}(n)$ is $(*)$ where $(*)$ is not $(\Box)_L$ or $(\Box)_R$. We assume that $(*)$ is applied to the formula φ in Γ_n or Δ_n and generates $\mathbb{P}(n+1)$, where (one of) the residual(s) is φ^*. Then we mark $\mathbb{P}(n+1)$ in two steps:
  1. Marking the residual formula and the dual of the principal formula:
     - If φ is unmarked in $\mathbb{P}(n)$ and the dual $D(\varphi)$ occurs on the other side of the sequent arrow in $\mathbb{P}(n)$, then we mark $D(\varphi)$ in $\mathbb{P}(n+1)$.
     - If φ is marked in $\mathbb{P}(n)$, φ^* does not occur unmarked on the same side as φ in $\mathbb{P}(n)$ and $D(\varphi^*)$ does not occur unmarked on the other side of the sequent arrow in $\mathbb{P}(n)$, then we mark φ^* in $\mathbb{P}(n+1)$.

2. Marking side-formulas:
   - If ψ occurs on the same side as φ such that $\psi \neq \varphi^*$ and ψ is marked, then we mark the occurrence of ψ in $\mathbb{P}(n+1)$.
   - If ψ occurs on the other side as φ such that $\psi \neq D(\varphi)$ and ψ is marked, then we mark the occurrence of ψ in $\mathbb{P}(n+1)$.

▷ If the rule applied at $\mathbb{P}(n)$ is $(\Box)_L$ or $(\Box)_R$ we do not mark any formula labelling $\mathbb{P}(n+1)$.

▷ No other formulas are marked.

We call such a proof a *marked proof* for $\Gamma \Rightarrow \Delta$.

Example 5.7.6. Consider the sequent $P \wedge Q, (R \wedge \neg P) \vee (R \wedge \neg Q) \Rightarrow \neg R \vee P, \neg R \vee Q$. Below is a proof for this sequent. We show how to step-wise mark the proof-tree. At the beginning, we have an unmarked proof t:

$$\dfrac{\dfrac{\dfrac{\dfrac{P, Q, R, \neg P \Rightarrow \neg R, P, \neg R \vee Q}{P \wedge Q, R, \neg P \Rightarrow \neg R, P, \neg R \vee Q}(\wedge)_L}{P \wedge Q, R, \neg P \Rightarrow \neg R \vee P, \neg R \vee Q}(\vee)_R}{P \wedge Q, R \wedge \neg P \Rightarrow \neg R \vee P, \neg R \vee Q}(\wedge)_L \qquad \dfrac{\dfrac{P, Q, R \wedge \neg Q \Rightarrow \neg R \vee P, \neg R, Q}{P \wedge Q, R \wedge \neg Q \Rightarrow \neg R \vee P, \neg R, Q}(\wedge)_L}{P \wedge Q, R \wedge \neg Q \Rightarrow \neg R \vee P, \neg R \vee Q}(\vee)_R}{P \wedge Q, (R \wedge \neg P) \vee (R \wedge \neg Q) \Rightarrow \neg R \vee P, \neg R \vee Q}(\vee)_L$$

In the first step we mark the root which implies that we do not add any marks to t. Let us focus on the left branch $\mathbb{P}$ first. The principal formula in $\mathbb{P}(0)$ is $(R \wedge \neg P) \vee (R \wedge \neg Q)$ which is unmarked and its dual does not occur on the right. Thus we do not mark any formula in $\mathbb{P}(1)$. Next, the principal formula of $\mathbb{P}(1)$ is $R \wedge \neg P$ which is unmarked, but its dual $\neg R \vee P$ occurs on the right. Hence we mark $\neg R \vee P$ in $\mathbb{P}(2)$. Next, the principal formula of $\mathbb{P}(2)$ is $\neg R \vee P$ which is marked. Notice that for both residuals of the application of $(\vee)_R$, namely $\neg R$ and P, their respective dual occurs unmarked on the left and so no formula is marked in $\mathbb{P}(3)$. Finally, the last step does not introduce any markings. Afterwards we also mark the right branch $\mathbb{P}'$. Here, the formula $R \wedge \neg Q$ is marked in $\mathbb{P}'(2)$ and its marking is not removed for the rest of the path. Thus we obtain the following marked proof:

$$\dfrac{\dfrac{\dfrac{\dfrac{P, Q, R, \neg P \Rightarrow \neg R, P, \neg R \vee Q}{P \wedge Q, R, \neg P \Rightarrow \neg R, P, \neg R \vee Q}(\wedge)_L}{P \wedge Q, R, \neg P \Rightarrow \neg \mathbf{R} \vee \mathbf{P}, \neg R \vee Q}(\vee)_R}{P \wedge Q, R \wedge \neg P \Rightarrow \neg R \vee P, \neg R \vee Q}(\wedge)_L \qquad \dfrac{\dfrac{P, Q, \mathbf{R} \wedge \neg \mathbf{Q} \Rightarrow \neg R \vee P, \neg R, Q}{P \wedge Q, \mathbf{R} \wedge \neg \mathbf{Q} \Rightarrow \neg R \vee P, \neg R, Q}(\wedge)_L}{P \wedge Q, R \wedge \neg Q \Rightarrow \neg R \vee P, \neg R \vee Q}(\vee)_R}{P \wedge Q, (R \wedge \neg P) \vee (R \wedge \neg Q) \Rightarrow \neg R \vee P, \neg R \vee Q}(\vee)_L$$

Having defined marked proofs, we now move on to define corresponding paths. Afterwards we give an example that illustrates what effect the marking has in building suitable proofs.

Definition 5.7.7. Let $t = (V, \rightarrow, \lambda)$ be a marked proof for $\Gamma \Rightarrow \Delta$ and let $t' = (V', \rightarrow', \lambda')$ be a pre-proof of $\Rightarrow D(\Gamma), \Delta$. Let $u \in V$ and $u' \in V'$ be nodes such that $\lambda(u) = (\Sigma \Rightarrow \Pi)$ and $\lambda'(u') = (\Rightarrow \Omega)$. We call u and u' *corresponding* - written $u \leftrightsquigarrow u'$ - if and only if

$$D(\Sigma - \{\varphi \in \Sigma | \varphi \text{ is marked }\}) \cup (\Pi - \{\varphi \in \Pi | \varphi \text{ is marked }\}) = \Omega$$

Let $\mathbb{P}$ be a path through t and $\mathbb{P}'$ a path through t'. We call $\mathbb{P}$ and $\mathbb{P}'$ *corresponding* - written $\mathbb{P} \leftrightsquigarrow \mathbb{P}'$ - if and only if the following conditions hold:

1. Either both paths are finite or both paths are infinite.
2. Every node in $\mathbb{P}$ corresponds to exactly one node in $\mathbb{P}'$ and conversely every node in $\mathbb{P}'$ corresponds to at least one node in $\mathbb{P}$.
3. If $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(m)$, then $n \geq m$.
4. If $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(m)$ and $\mathbb{P}(k) \leftrightsquigarrow \mathbb{P}'(l)$ where $n < k$, then $m \leq l$.

Example 5.7.8. Recall the marked proof for $P \wedge Q, (R \wedge \neg P) \vee (R \wedge \neg Q) \Rightarrow \neg R \vee P, \neg R \vee Q$ from the previous example. We use this proof to build a proof for

$$\Rightarrow D(P \wedge Q), D((R \wedge \neg P) \vee (R \wedge \neg Q)), \neg R \vee P, \neg R \vee Q$$

where each path corresponds to a path in the marked proof above. We do this by applying the same (or the dual) rules to the same (or the dual) formulas in the same order, where whenever the principal formula is marked, we ignore that step. Thus we obtain the following prooftree:

$$\dfrac{\dfrac{\dfrac{\Rightarrow \neg P, \neg Q, \neg R, P, \neg R \vee Q}{\Rightarrow \neg P \vee \neg Q, \neg R, P, \neg R \vee Q}\,(\vee)_R}{\Rightarrow \neg P \vee \neg Q, \neg R \vee P, \neg R \vee Q}\,(\vee)_R \qquad \dfrac{\dfrac{\Rightarrow \neg P, \neg Q, \neg R \vee P, \neg R, Q}{\Rightarrow \neg P \vee \neg Q, \neg R \vee P, \neg R, Q,}\,(\vee)_R}{\Rightarrow \neg P \vee \neg Q, \neg R \vee P, \neg R \vee Q}\,(\vee)_R}{\Rightarrow \neg P \vee \neg Q, (\neg R \vee P) \wedge (\neg R \vee Q), \neg R \vee P, \neg R \vee Q}\,(\wedge)_R$$

Notice that the left path corresponds to the left path of the marked proof and the right path corresponds to the right path of the marked proof. Moreover, notice that we can build this prooftree entirely syntactical, given that we already have the marked proof. There is no reasoning involved, but only applying an algorithm that tells us which rules to apply to which formulas on the basis of the marked prooftree. This example can be generalized to arbitrary prooftrees which establishes the converse of theorem 5.7.3.

Theorem 5.7.9. Let $\Gamma \Rightarrow \Delta$ be a sequent. If $\mathbf{2DT} \vdash \Gamma \Rightarrow \Delta$, then $\mathbf{2DT} \vdash \Rightarrow D(\Gamma), \Delta$.

Proof. Let $\Gamma \Rightarrow \Delta$ be a sequent and let $t = (V, \rightarrow, \lambda)$ be a marked proof for $\Gamma \Rightarrow \Delta$. We show how to construct a proof $t' = (V', \rightarrow', \lambda')$ for $\Rightarrow D(\Gamma), \Delta$.

▷ The root $r_{t'}$ of t' is labelled by $\Rightarrow D(\Gamma), \Delta$. Notice that $r_t \leftrightsquigarrow r_{t'}$.

▷ Suppose we have constructed the path $\mathbb{P}'(0)...\mathbb{P}'(m)$ which corresponds to the initial segment of the path $\mathbb{P}(0)...\mathbb{P}(n)$ through t where $m \leq n$. In particular we assume that $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(m)$. We show how to extend $\mathbb{P}'$:

- Suppose $\lambda(\mathbb{P}(n)) = (\Gamma_n, \varphi \wedge \psi \Rightarrow \Delta_n)$, the rule applied at $\mathbb{P}(n)$ is $(\wedge)_L$ and $\mathbb{P}(n+1)$ is labelled by $\Gamma_n, \varphi, \psi \Rightarrow \Delta_n$. We distinguish two cases:
  1. Suppose $\varphi \wedge \psi$ is marked. So there has been an application of $(\vee)_R$ in an earlier node of $\mathbb{P}$ to $D(\varphi \wedge \psi)$ on the right, where $\varphi \wedge \psi$ was already present on the left. If φ occurs unmarked on the left in $\mathbb{P}(n+1)$, then the formula $D(\varphi)$ occurs unmarked on the right side of $\mathbb{P}(n)$ or φ occurs unmarked on the left side of $\mathbb{P}(n)$. In both cases, since $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(m)$, we have that $D(\varphi)$ labels the right side of $\mathbb{P}'(m)$. The case where ψ is unmarked in $\mathbb{P}(n+1)$ is identical. If φ occurs marked in $\mathbb{P}(n+1)$, then either $D(\varphi)$ does not occur on the right side of $\mathbb{P}(n)$ or $D(\varphi)$ occurs marked on the right side of $\mathbb{P}(n)$ and φ does not occur unmarked on the left side of $\mathbb{P}(n)$. Since $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(m)$, the right side of $\mathbb{P}'(m)$ is not labelled by $D(\varphi)$ and therefore $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m)$. The case where ψ is marked is identical. Thus in all cases $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m)$ and we do not apply any rules to $\mathbb{P}'(m)$. Notice that $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(0)...\mathbb{P}'(m)$.
  2. Suppose $\varphi \wedge \psi$ is unmarked . Then both φ and ψ labelling $\mathbb{P}(n+1)$ on the left side are unmarked as well. Moreover $D(\varphi) \vee D(\psi)$ labels the right side of $\mathbb{P}'(m)$. Apply the rule $(\vee)_R$ to generate a new node u labelled by $D(\varphi)$ and by $D(\psi)$ on the right and let $\mathbb{P}'(m+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+1)$ and $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(0)...\mathbb{P}'(m)\mathbb{P}'(m+1)$.
- Suppose $\lambda(\mathbb{P}(n)) = (\Gamma_n \Rightarrow \varphi \wedge \psi, \Delta_n)$, the rule applied is $(\wedge)_R$ which generates two new nodes u and v labelled by $\Gamma_n \Rightarrow \varphi, \Delta_n$ and $\Gamma_n \Rightarrow \psi, \Delta_n$ respectively. We distinguish two cases:
  1. Suppose $\varphi \wedge \psi$ is marked. Then there has been an earlier application in $\mathbb{P}$ of the $(\vee)_L$-rule to $D(\varphi) \vee D(\psi)$ on the left, where $\varphi \wedge \psi$ was already present on the right which split up the path in two. Suppose without loss of generality that we chose $D(\varphi)$. Then we assume that $\mathbb{P}(n+1) = u$ (if not, we simply consider the path which goes through u and which is identical to $\mathbb{P}$ up to the node $\mathbb{P}(n)$). In case the occurrence of φ that labels the right side of $\mathbb{P}(n+1)$ is unmarked, we have that the formula $D(\varphi)$ occurs unmarked on the left side of $\mathbb{P}(n)$ or φ occurs unmarked on the right side of $\mathbb{P}(n)$. In both cases, since $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(m)$, we have that φ occurs on the right of $\mathbb{P}'(m)$ and therefore $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m)$. If φ is marked, then φ does not occur unmarked on the right side of $\mathbb{P}(n)$ and $D(\varphi)$ does not occur unmarked on the left of $\mathbb{P}(n)$. Thus, since $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(m)$, we have that φ does not occur on the right side of $\mathbb{P}'(m)$ and so $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m)$. Therefore we do not apply any rules to $\mathbb{P}'(m)$ and we notice that $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(0)...\mathbb{P}'(m)$.

2. Suppose $\varphi \wedge \psi$ is unmarked and without loss of generality $\mathbb{P}(n+1) = u$. Then $\varphi \wedge \psi$ labels the right side of $\lambda'(\mathbb{P}'(m))$ and we apply the rule $(\wedge)_R$ to generate two nodes u and v labelled on the right by φ and ψ respectively. Let $\mathbb{P}'(m+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+1)$ and $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(0)...\mathbb{P}'(m)\mathbb{P}'(m+1)$.

- Suppose $\lambda(\mathbb{P}(n)) = (\Gamma_n, \varphi \vee \psi \Rightarrow \Delta_n)$ and the rule applied is $(\vee)_L$ which generates two nodes u and v labelled by $\Gamma_n, \varphi \Rightarrow \Delta_n$ and $\Gamma_n, \psi \Rightarrow \Delta_n$ respectively. This case is symmetric to the previous one.
- Suppose $\lambda(\mathbb{P}(n)) = (\Gamma_n \Rightarrow \varphi \vee \psi, \Delta_n)$, the rule applied is $(\vee)_R$ and $\mathbb{P}(n+1)$ is labelled by $\Gamma_n \Rightarrow \varphi, \psi, \Delta_n$. This case is symmetric to the first one.
- Suppose $\lambda(\mathbb{P}(n)) = \Sigma, \Box\Gamma_n, \Diamond\varphi \Rightarrow \Diamond\Delta_n, \Omega$, the rule applied is $(\Box)_L$ and $\mathbb{P}(n+1)$ is labelled by $\Gamma_n, \varphi \Rightarrow \Delta_n$. We distinguish two cases:
  1. Suppose some formula $\psi \in \Box\Gamma_n \cup \{\Diamond\varphi\} \cup \Diamond\Delta_n$ is marked. This implies that there has been an earlier node in the path which was labelled by both ψ on one side and its dual $D(\psi)$ on the other side, and the rule applied to that node decomposed $D(\psi)$. But this rule can only have been $(\Box)_L$ or $(\Box)_R$, which implies that also ψ has been decomposed in that step, which is a contradiction. Therefore this case cannot happen.
  2. Suppose every formula in $\Box\Gamma_n \cup \{\Diamond\varphi\} \cup \Diamond\Delta_n$ is unmarked. Then

$$\lambda'(\mathbb{P}'(m)) = (\Rightarrow \Diamond D(\Gamma_n), \Box D(\varphi), \Diamond\Delta_n, \Theta, D(\Phi))$$

     for some set of formulas $\Theta \subseteq \Omega$ and some set of formulas $\Phi \subseteq \Sigma$. We apply the rule $(\Box)_R$ to generate a new node u which is labelled by $\Rightarrow D(\Gamma_n), D(\varphi), \Delta_n$ and let $\mathbb{P}'(m+1) = u$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+1)$ and $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(0)...\mathbb{P}'(m)\mathbb{P}'(m+1)$.
- Suppose $\lambda(\mathbb{P}(n)) = \Sigma, \Box\Gamma_n \Rightarrow \Box\varphi, \Diamond\Delta_n, \Omega$, the rule applied is $(\Box)_R$ and $\mathbb{P}(n+1)$ is labelled by $\Gamma_n \Rightarrow \varphi, \Delta_n$. This case is symmetric to the previous one.
- Suppose $\lambda(\mathbb{P}(n)) = (\Gamma_n, \sigma Z.\varphi(Z) \Rightarrow \Delta_n)$ for $\sigma \in \{\mu, \nu\}$, the rule applied is $(\sigma)_L$ and $\mathbb{P}(n+1)$ is labelled by $\Gamma_n, Z \Rightarrow \Delta_n$. Again we distinguish two cases:
  1. Suppose $\sigma Z.\varphi(Z)$ is marked. Then by similar reasoning as in the first step we have that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m)$ and we do not extend $\mathbb{P}'$.
  2. Suppose $\sigma Z.\varphi(Z)$ is unmarked. Then $D(\sigma Z.\varphi(Z))$ occurs on the right side of the sequent which labels $\mathbb{P}'(m)$ and we apply the dual rule of $(\sigma)_L$ to extend $\mathbb{P}'$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+1)$.
- The cases where $\lambda(\mathbb{P}(n))$ is labelled by $\sigma Z.\varphi(Z)$ on the right or by Z on the left or on the right are similar to the case before.

We have now shown how to construct t'. First of all, notice that t' is a finite branching tree and its root is labelled by $\Rightarrow D(\Gamma), \Delta$. Moreover, the tree is generated by **2DT**-rules. We have shown that each path through t' corresponds to a path through t (notice that the converse is not true in general). Now suppose u is a leaf of t'. Let $\mathbb{P}' = \mathbb{P}'(0)...\mathbb{P}'(m) = u$ be the finite path that leads to u. There exists a corresponding path $\mathbb{P} = \mathbb{P}(0)...\mathbb{P}(n)$ through t where $n \geq m$ and $\mathbb{P}(n)$ is a leaf as well. Moreover $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(m)$. Since $\mathbb{P}(n)$ is a leaf and t a proof we have that $\mathbb{P}(n)$ is an axiomatic leaf. Finally notice that literals are always unmarked. This directly implies that u is an axiomatic leaf as well. Now suppose that $\mathbb{P}'$ is an infinite path through t' and let $\mathbb{P}$ be its corresponding infinite path through t. Since t is a proof we have that $\mathbb{P}$ contains an infinite μ-trace on the left or an infinite ν-trace on the right which we denote by tr. Let tr' denote its corresponding ν-trace on the right in $\mathbb{P}'$. Suppose towards a contradiction that tr' is finite. This is only possible, if from some node on the trace tr in $\mathbb{P}$ is always marked. Recall that every formula is guarded. Hence we have that every infinite path passes through a box-rule infinitely often and we have already seen that the formulas surviving a box-rule are always unmarked. Thus, as soon as we reach the node in $\mathbb{P}$ from where on tr is always marked, the trace tr will end at the next application of a box-rule, which contradicts the assumption that tr is infinite. Therefore tr' is infinite and so $\mathbb{P}'$ has an infinite ν-trace on the right. We conclude that t' is a proof for $\Rightarrow D(\Gamma), \Delta$. □

Combining theorem 5.7.3 and theorem 5.7.9 yields the following corollary:

Corollary 5.7.10. Let $\Gamma \Rightarrow \Delta$ be a sequent. Then the following holds:

$$\mathbf{2DT} \vdash \Gamma \Rightarrow \Delta \;\Leftrightarrow\; \mathbf{2DT} \vdash \Rightarrow D(\Gamma), \Delta$$

Theorem 5.7.11 (Soundness and completeness of **2DT**)**.** Let $\Gamma \Rightarrow \Delta$ be a sequent. Then the following holds:

$$\mathbf{2DT} \vdash \Gamma \Rightarrow \Delta \text{ if and only if } \Gamma \Rightarrow \Delta \text{ is valid.}$$

Proof. Let $\Gamma \Rightarrow \Delta$ be a sequent. Then we have that

$$\begin{array}{rcll} \mathbf{2DT} \vdash \Gamma \Rightarrow \Delta & \Leftrightarrow & \mathbf{2DT} \vdash \Rightarrow D(\Gamma), \Delta & \text{(Corollary 5.7.10)} \\ & \Leftrightarrow & \mathbf{DT}' \vdash D(\Gamma), \Delta & \text{(Theorem 5.7.1)} \\ & \Leftrightarrow & D(\Gamma), \Delta \text{ is valid} & \text{(Theorem 5.5.4 and Theorem 5.5.1)} \\ & \Leftrightarrow & \bigvee D(\Gamma) \vee \bigvee \Delta \text{ is valid} & \text{(by definition)} \\ & \Leftrightarrow & \bigwedge \Gamma \rightarrow \bigvee \Delta \text{ is valid} & \text{(by propositional reasoning)} \\ & \Leftrightarrow & \Gamma \Rightarrow \Delta \text{ is valid} & \text{(by definition)} \end{array}$$ □

Chapter 6

Craig interpolation

6.1 Introduction

A logic $\mathcal{L}$ is said to have Craig interpolation, if whenever an $\mathcal{L}$-implication $\varphi \to \psi$ is valid, there exists a formula γ which uses only non-logical symbols that occur in both φ and ψ, such that $\varphi \to \gamma$ and $\gamma \to \psi$ are valid. The formula γ is called an interpolant for $\varphi \to \psi$. Craig interpolation was first stated and proved for classical first-order logic by Craig in 1957. Interpolation results have been established for many different non-classical logical systems ever since. For a concise overview we refer to [6] by D'Agostino. There are many reasons why it is interesting to study Craig interpolation. As a start, Craig interpolation is used as a tool to prove results in mathematical logic such as the Beth Theorem or the Łos-Tarski-Theorem. Moreover, interpolation is applied to computer science in areas such as software design, database theory and model checking. While many non-classical logics enjoy Craig interpolation, it is not a common property among fixed point logics. Famous systems such as LTL, CTL or CTL* all fail to have interpolation [6]. However, interpolation results have been established for the modal mu-calculus. Uniform interpolation was established by D'Agostino and Hollenberg in 2000 [7]. Most recently, Afshari and Leigh established Lyndon interpolation [3]. Both uniform and Lyndon interpolation are stronger interpolation properties than Craig interpolation. As this chapter deals with Craig interpolation, we mean by interpolation for the rest of the chapter always Craig interpolation. There are different approaches for establishing that a logic enjoys interpolation. Interpolation for intuitionistic logic for instance can be established using algebraic methods. Instead of proving Craig interpolation directly, one establishes it by proving that the variety of Heyting algebras enjoys amalgamation, which is the algebraic counterpart of interpolation. There are also model theoretic approaches which use Robinson's consistency theorem. Another approach uses methods from proof theory. Given a cut-free sequent calculus and a derivation of $\varphi \to \psi$, one constructs an interpolant for $\varphi \to \psi$ by starting to interpolate the leafs of the derivation tree. Afterwards, one shows how to construct interpolants for the conclusion of a rule, given that interpolants for its premises have already been constructed.

The proof theoretic approach therefore results in a constructive proof of Craig interpolation. In this chapter we establish that the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$ enjoys interpolation, by following the proof theoretic approach. That is, we show how to construct interpolants from given proofs. The main problem is that every proof system for the modal mu-calculus which we have considered so far allows infinite proof trees. The proof theoretic approach however does not work on an infinite tree, as it relies on starting the construction of the interpolant in the leafs. Afshari and Leigh provided a solution to this problem by turning towards circular proof systems for the modal mu-calculus [3]. As a circular proof is finite, one can build interpolants using the standard proof-theoretic approach. We take a similar route and define a circular proof system for $\Sigma_1^\mu \cup \Pi_1^\mu$. As we restrict to the first level of the alternation depth hierarchy, we do not have to deal with syntactic fixed point alternation of degree higher than 1, which implies that the circular calculus turns out to be much simpler than what is needed for the whole modal mu-calculus. This is due to similar reasons as were explained in chapter 4 when we introduced circular tableaux. The system we work with is obtained from the two-sided calculus **2DT** by pruning the infinitary proof trees at suitable nodes. The sequent arrow $\Rightarrow$ is thereby a natural candidate to interpret the implication $\rightarrow$ in the definition of Craig interpolation. After we define this circular calculus and proof its soundness and completeness with respect to $\Sigma_1^\mu \cup \Pi_1^\mu$ in section 6.2, we show how to use it to establish Craig interpolation. We first show that the fragments Σ_1^μ and Π_1^μ enjoy Craig interpolation in section 6.3. The constructed Craig interpolant for these two fragments turns out to be optimal, in the sense that the interpolant constructed in the proof for Σ_1^μ belongs itself to Σ_1^μ, while the interpolant constructed for Π_1^μ belongs to Π_1^μ. We then establish that $\Sigma_1^\mu \cup \Pi_1^\mu$ enjoys Craig interpolation by combining the previous results in section 6.4. The constructed interpolant however does in general not lie in $\Sigma_1^\mu \cup \Pi_1^\mu$, but has arbitrary fixed point alternation depth. We show in the last section 6.5 of this chapter how and to what extent this result can be optimized.

6.2 The circular sequent calculus C2DT

This section introduces the circular and two-sided sequent calculus **C2DT** for $\Sigma_1^\mu \cup \Pi_1^\mu$. Similar to circular tableaux introduced in chapter 4, a circular proof is a finite tree with loops that unravels into an infinitary and regular proof. As we consider two-sided sequents, the circular proofs unravel into infinitary **2DT**-proofs. We first define the circular sequent calculus **C2DT**. Afterwards we prove its soundness and completeness with respect to the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$. As for circular tableaux, the definition of a circular proof as presented here is not sound for formulas which are in a higher level of the alternation hierarchy. Indeed, our definition relies on the lack of fixed point alternation. From now on, given a sequent $\Gamma \Rightarrow \Delta$, we assume that both Γ and Δ are subsets of $\Sigma_1^\mu \cup \Pi_1^\mu$.

Definition 6.2.1. The circular sequent calculus **C2DT** consists of the same axioms and rules as the two-sided sequent calculus **2DT** and is depicted in Table 6.1.

Table 6.1: The circular sequent calculus **C2DT**

$$\frac{}{\Gamma, P \Rightarrow P, \Delta}\ (\mathrm{Ax})_1 \qquad \frac{}{\Gamma \Rightarrow P, \neg P, \Delta}\ (\mathrm{Ax})_2$$

$$\frac{}{\Gamma, \neg P \Rightarrow \neg P, \Delta}\ (\mathrm{Ax})_3 \qquad \frac{}{\Gamma, P, \neg P \Rightarrow \Delta}\ (\mathrm{Ax})_4$$

$$\frac{\Gamma, \varphi_0, \varphi_1 \Rightarrow \Delta}{\Gamma, \varphi_0 \wedge \varphi_1 \Rightarrow \Delta}\ (\wedge)_L \qquad \frac{\Gamma \Rightarrow \varphi_0, \Delta \quad \Gamma \Rightarrow \varphi_1, \Delta}{\Gamma \Rightarrow \varphi_0 \wedge \varphi_1, \Delta}\ (\wedge)_R$$

$$\frac{\Gamma, \varphi_0 \Rightarrow \Delta \quad \Gamma, \varphi_1 \Rightarrow \Delta}{\Gamma, \varphi_0 \vee \varphi_1 \Rightarrow \Delta}\ (\vee)_L \qquad \frac{\Gamma \Rightarrow \varphi_0, \varphi_1, \Delta}{\Gamma \Rightarrow \varphi_0 \vee \varphi_1, \Delta}\ (\vee)_R$$

$$\frac{\Gamma, \varphi \Rightarrow \Delta}{\Theta, \Box\Gamma, \Diamond\varphi \Rightarrow \Diamond\Delta, \Sigma}\ (\Box)_L \qquad \frac{\Gamma \Rightarrow \varphi, \Delta}{\Theta, \Box\Gamma \Rightarrow \Box\varphi, \Diamond\Delta, \Sigma}\ (\Box)_R$$

$$\frac{\Gamma, Z \Rightarrow \Delta}{\Gamma, \mu Z.\varphi(Z) \Rightarrow \Delta}\ (\mu)_L \qquad \frac{\Gamma \Rightarrow Z, \Delta}{\Gamma \Rightarrow \mu Z.\varphi(Z), \Delta}\ (\mu)_R$$

$$\frac{\Gamma, Z \Rightarrow \Delta}{\Gamma, \nu Z.\varphi(Z) \Rightarrow \Delta}\ (\nu)_L \qquad \frac{\Gamma \Rightarrow Z, \Delta}{\Gamma \Rightarrow \nu Z.\varphi(Z), \Delta}\ (\nu)_R$$

$$\frac{\Gamma, \varphi(Z) \Rightarrow \Delta}{\Gamma, Z \Rightarrow \Delta}\ (Z)_L \qquad \frac{\Gamma \Rightarrow \varphi(Z), \Delta}{\Gamma \Rightarrow Z, \Delta}\ (Z)_R$$

Definition 6.2.2. A *circular pre-proof* for $\Gamma \Rightarrow \Delta$ is a finite tree $t = (V, \to, \lambda)$ with root r_t which is generated by the rules of **C2DT** such that:

1. $\lambda(r_t) = \Gamma \Rightarrow \Delta$
2. every leaf $u \in V$ is either labelled by an axiom or by a sequent of the form $\Theta, \Box\Gamma \Rightarrow \Diamond\Delta, \Sigma$ where $\Theta \subseteq Lit$ and $\Sigma \subseteq Lit$ or by a sequent $\Pi \Rightarrow \Omega$, such that there exists a distinguished node $u' \in V$ from which u is reachable and $\lambda(u') = \lambda(u)$. We call u' the *associated node* of u.

Observe that circular pre-proofs are finite trees. The second condition ensures that pre-proofs are maximal. If u is a leaf, then there are three possibilities:

1. The leaf u is an axiom
2. The leaf u is a node where no more rules can be applied

3. The leaf u has an associated node u' that occurs earlier in the branch leading to u such that u can be identified with u'.

As for circular tableaux, leafs in the third case are used to introduce loops into the prooftree. Leafs in case one and two are called leafs of *type 1* and leafs in case three are called leafs of *type 2*. We define a path through a pre-proof as follows:

Definition 6.2.3. Let $t = (V, \rightarrow, \lambda)$ be a circular pre-proof. A *path* $\mathbb{P}$ through t is a (possibly infinite) sequence of nodes $\mathbb{P}(0)\mathbb{P}(1)\mathbb{P}(2)...$ with $\mathbb{P}(0) = r_t$ such that for all $i \in \omega$:

1. If $\mathbb{P}(i)$ is not a leaf, then $\mathbb{P}(i) \rightarrow \mathbb{P}(i+1)$.
2. If $\mathbb{P}(i)$ is a leaf of type 1, then the path ends at $\mathbb{P}(i)$.
3. If $\mathbb{P}(i)$ is a leaf of type 2 and $j < i$ such that $\mathbb{P}(j)$ is the associated node of $\mathbb{P}(i)$, then $\mathbb{P}(j) \rightarrow \mathbb{P}(i+1)$.

Traces through a path are defined as in definition 3.3.8.

Definition 6.2.4. A *circular proof* for $\Gamma \Rightarrow \Delta$ is a circular pre-proof $t = (V, \rightarrow, \lambda)$ for $\Gamma \Rightarrow \Delta$, such that every leaf of type 1 is labelled by an axiom and for every leaf u of type 2 labelled by $\Pi \Rightarrow \Omega$ with associated node u' the following holds:

1. between u' and u there is an application of the rule $(\Box)_L$ or $(\Box)_R$ and
2. there exists a formula $\varphi \in \Pi$ which belongs to a trace that starts in a Σ_1^μ-formula or there exists a formula $\varphi \in \Omega$ which belongs to a trace that starts in a Π_1^μ-formula. The formula φ is called the *distinguished formula of u*.

We write $\mathbf{C2DT} \vdash \Gamma \Rightarrow \Delta$ if and only if there exists a **C2DT**-proof for $\Gamma \Rightarrow \Delta$. Leafs of type 1 which are labelled by an axiom are called *axiomatic leafs*. Leafs of type 2 that fulfil the two conditions above are called *non-axiomatic leafs*.

Remark 6.2.5. We assume without loss of generality that the distinguished formula φ of a non-axiomatic leaf u belongs to the same trace as the occurrence of φ in the associated node u'. If this is not the case, then the occurrence of φ labelling u is a sub-formula of some formula ψ, which labels both u' and u and which is decomposed and regenerated between u' and u, such that φ is freed in the process. However, this implies that both occurrences of ψ belong to the same trace. Moreover, since we only consider the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$ and φ is a subformula of ψ, the formula ψ belongs to a trace that starts in a Σ_1^μ-formula on the left or a Π_1^μ-formula on the right. Therefore we can simply choose the distinguished formula to be ψ.

Observe that circular pre-proofs for a sequent are not unique. There are no restrictions on which suitable repetition should be chosen to serve as non-axiomatic leaf. Therefore it is possible for a sequent to have infinitely many different circular pre-proofs. Recall the discussion in chapter 4 about why the restriction to the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$ matters for the soundness of the circular tableaux system. Given a circular tableaux for a sequent that contains formulas with a greater alternation depth, we noticed that we need more refined conditions for non-axiomatic leafs to ensure that unfolding the circular tableau leads to a tableau. The same goes for circular proofs. The restriction to the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$ is essential for the system's soundness.

6.2.1 Soundness and completeness of C2DT

In order to establish that **C2DT** is sound and complete with respect to the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$, we use the soundness and completeness of **2DT**. For soundness, we unravel a circular proof into a **2DT**-proof. For completeness, we first show that given a **2DT**-proof t, every infinite path through t contains a repetition of a suitable form and then we show how to prune a **2DT**-proof into a circular proof.

Theorem 6.2.6 (Soundness of **C2DT** with respect to $\Sigma_1^\mu \cup \Pi_1^\mu$)**.** Let $\Gamma \Rightarrow \Delta$ be a sequent such that $\Gamma, \Delta \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$. If $\mathbf{C2DT} \vdash \Gamma \Rightarrow \Delta$, then $\Gamma \Rightarrow \Delta$ is valid.

Proof. Suppose that $\Gamma, \Delta \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ and $t = (V, \to, \lambda)$ is a circular proof of $\Gamma \Rightarrow \Delta$. First of all, notice that if every leaf of t is axiomatic, then t is also a **2DT**-proof of $\Gamma \Rightarrow \Delta$. Soundness of **2DT** then implies that $\Gamma \Rightarrow \Delta$ is valid. Thus, suppose that there are $n \geq 1$ non-axiomatic leafs in t. The idea is to unravel t over its non-axiomatic leafs into an infinite tree t'. That is, we define t' in stages, where $t'_0 = t$ and t'_{s+1} is t'_s where for each leaf of t'_s corresponding to a non-axiomatic leaf u of t we plug the sub-tree of t rooted at the successor of u's associated node on top of it. The tree t' is then defined to be the limit of this construction. Notice that t' is an infinite tree generated by the rules of **2DT** and whose root is labelled by $\Gamma \Rightarrow \Delta$. By construction every leaf of t' corresponds to an axiomatic leaf of t and is thus axiomatic itself. Moreover, every infinite path $\mathbb{P}'$ through t' corresponds to an infinite path through t. It therefore suffices to show that every infinite path through t has an infinite μ-trace on the left or an infinite ν-trace on the right. Every infinite path through t passes through some non-axiomatic leaf u with associated node u' infinitely often. Suppose without loss of generality that the distinguished formula φ of u occurs on the left. By definition φ belongs to a trace that starts in a Σ_1^μ-formula. Hence, there are infinitely many nodes of $\mathbb{P}$ that are labelled by Σ_1^μ-formulas on the left. This implies that every node of $\mathbb{P}$ is labelled by Σ_1^μ-formulas on the left. If this was not the case, then there would exist $n \in \omega$ such that $\mathbb{P}(n)$ was labelled by Π_1^μ-formulas on the left only. But since we are working in the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$, there is no way to reintroduce Σ_1^μ-formulas on the left. Therefore for each $m > n$, the node $\mathbb{P}(m)$ would also be labelled by Π_1^μ-formulas on the left only. This contradicts the assumption that there are infinitely many nodes labelled by φ on the left. Since $\mathbb{P}$ is infinite and each node of $\mathbb{P}$ is labelled by Σ_1^μ-formulas on the left, there exists - by lemma 4.2.3 - an infinite trace on the left, which is a μ-trace. The case where the distinguished formula of u occurs on the right is identical. Therefore, every infinite path through t has an infinite μ-trace on the left or an infinite ν-trace on the right and so does every path through t' which implies that t' is a **2DT**-proof of $\Gamma \Rightarrow \Delta$. Finally, since **2DT** is sound, we conclude that $\Gamma \Rightarrow \Delta$ is valid and thus that **C2DT** is sound. $\square$

For proving completeness, we use the completeness of **2DT**. Suppose $\Gamma \Rightarrow \Delta$ is valid. Then it has a **2DT**-proof $t = (V, \to, \lambda)$. We search each infinite branch of t for suitable repetitions and prune it there to obtain a finite tree, which serves as the circular proof. For that we first have to establish that suitable repetitions exist in each infinite branch, for which in turn we require the following proposition:

Proposition 6.2.7. Every infinite path in a **2DT**-proof passes through a modality rule infinitely often.

The proof is a easy generalization of the proof of proposition 3.3.7.

Lemma 6.2.8. Let $t = (V, \to, \lambda)$ be a **2DT**-proof for $\Gamma \Rightarrow \Delta$ where $\Gamma, \Delta \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ and let $\mathbb{P}$ be an infinite path through t. There exist $j < i$ such that $\lambda(P(j)) = \lambda(P(i))$, there is an application of $(\Box)_L$ or $(\Box)_R$ between $\mathbb{P}(j)$ and $\mathbb{P}(i)$ and there exists $\varphi \in \lambda(\mathbb{P}(i))$ which either occurs on the left and belongs to a trace starting in a Σ_1^μ-formula or occurs on the right and belongs to a trace starting in a Π_1^μ-formula.

Proof. The root of t is by assumption labelled by $\Gamma \Rightarrow \Delta$. Therefore every sequent $\Pi \Rightarrow \Omega$ labelling some node $\mathbb{P}(n)$ has the property that $\Pi \subseteq Sub(\Gamma)$ and $\Omega \subseteq Sub(\Delta)$. Hence, there are only finitely many different sequents that can label nodes in $\mathbb{P}$. Since $\mathbb{P}$ is infinite, this implies that there exists a sequent $\Pi \Rightarrow \Omega$ that labels infinitely many nodes in $\mathbb{P}$. Since $\mathbb{P}$ passes through a modality rule infinitely often, there exists $j < i$, such that $\lambda(\mathbb{P}(j)) = \lambda(\mathbb{P}(i))$ and there is an application of a modality rule in between. Finally, since t is a proof we have that $\mathbb{P}$ has an infinite ν-trace on the right or an infinite μ-trace on the left. So there exists a formula φ labelling the left or the right side of $\mathbb{P}(i)$ which belongs to a μ- or a ν-trace. Since we are only considering the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$, this implies that φ belongs to a trace that starts in a Σ_1^μ-formula or in a Π_1^μ-formula. $\square$

Theorem 6.2.9 (Completeness of **C2DT** with respect to $\Sigma_1^\mu \cup \Pi_1^\mu$)**.** Let $\Gamma \Rightarrow \Delta$ be a sequent such that $\Gamma, \Delta \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$. If $\Gamma \Rightarrow \Delta$ is valid, then **C2DT** $\vdash \Gamma \Rightarrow \Delta$.

Proof. Suppose $\Gamma, \Delta \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ such that $\Gamma \Rightarrow \Delta$ is valid. By completeness of **2DT** there exists a proof $t = (V, \to, \lambda)$ of $\Gamma \Rightarrow \Delta$. First of all notice that if t is finite, then t is also a **C2DT**-proof. So suppose t is infinite and let $\mathbb{P}$ be an infinite path through t. By lemma 6.2.8 there exists $j < i$, such that $\lambda(\mathbb{P}(j)) = \lambda(\mathbb{P}(i))$, there is an application of a modality rule in between and there exists $\varphi \in \lambda(\mathbb{P}(i))$ which occurs on the left and belongs to a trace starting in a Σ_1^μ-formula or which occurs on the right and belongs to a trace starting in a Π_1^μ-formula. Prune the path $\mathbb{P}$ at this node. The pruned path is finite and $\mathbb{P}(i)$ is by definition a non-axiomatic leaf with associated node $\mathbb{P}(j)$. Thus, after pruning each infinite path of

t at a suitable repetition, we are left with a tree $t' = (V', \rightarrow', \lambda')$ consisting of finite paths only. König's Lemma implies that t' is finite. Moreover each leaf is either axiomatic or non-axiomatic. As the root of t' is labelled by $\Gamma \Rightarrow \Delta$ and t' is generated by **C2DT**-rules, we conclude that t' is a **C2DT**-proof of $\Gamma \Rightarrow \Delta$. Therefore **C2DT** is complete with respect to $\Sigma_1^\mu \cup \Pi_1^\mu$. □

6.3 Craig interpolation for Σ_1^μ and Π_1^μ

This section establishes the Craig interpolation property for Σ_1^μ- and Π_1^μ-implications, using the circular sequent calculus **C2DT**. Recall that $Sub(\varphi)$ is the set of all subformulas of the formula φ and given a finite set of L_μ-formulas Γ, $Sub(\Gamma) = \bigcup_{\varphi\in\Gamma} Sub(\varphi)$. Let

$$At(\Gamma) := \{P \in Prop | P \in Sub(\Gamma) \text{ or } \neg P \in Sub(\Gamma)\}$$

Definition 6.3.1. Let Γ, Δ be finite sets of L_μ-formulas. The *common language* $L(\Gamma) \cap L(\Delta)$ of Γ and Δ is the sub-language of L_μ consisting of the same set of variables, the same logical connectives and the same modal and fixed point operators as L_μ, as well as the set of atomic propositions $At(\Gamma) \cap At(\Delta) \subseteq Prop$.

Formulas of $L(\Gamma) \cap L(\Delta)$ in guarded normal form are defined as for $\mathcal{L}_\mu$. Observe that every $L(\Gamma) \cap L(\Delta)$-formula is therefore a L_μ-formula. If $At(\Gamma) \cap At(\Delta) = \emptyset$, we say that that the common language of Γ and Δ is *empty*, written $L(\Gamma) \cap L(\Delta) = \emptyset$.

Theorem 6.3.2 (Craig interpolation for Σ_1^μ)**.** Let $\Gamma, \Delta \subseteq \Sigma_1^\mu$. If $\Gamma \Rightarrow \Delta$ is valid and $\mathcal{L}(\Gamma) \cap \mathcal{L}(\Delta) \neq \emptyset$, then there exists an $\mathcal{L}(\Gamma) \cap \mathcal{L}(\Delta)$-formula γ such that $\Gamma \Rightarrow \gamma$ and $\gamma \Rightarrow \Delta$ are valid.

As mentioned in the introduction of this chapter, we follow the proof theoretic approach to establish Craig interpolation, using the circular sequent calculus **C2DT**. Suppose $\Gamma \Rightarrow \Delta$ is valid and their common language non-empty. By theorem 6.2.9 there exists a circular proof $t = (V, \rightarrow, \lambda)$ of $\Gamma \Rightarrow \Delta$. The interpolant for $\Gamma \Rightarrow \Delta$ is built by induction on t. In the base case the leaves of t are interpolated. This is done by defining a formula γ_0 for each leaf (axiomatic and non-axiomatic) which belongs to the common language of Γ and Δ and is called a *pre-interpolant*. In the case of an axiomatic leaf u labelled by $\Pi \Rightarrow \Omega$, the pre-interpolant γ_0 is constructed in such a way, that

$$\mathbf{C2DT} \vdash \Pi \Rightarrow \gamma_0 \text{ and } \mathbf{C2DT} \vdash \gamma_0 \Rightarrow \Omega \tag{6.1}$$

which implies that γ_0 is almost an interpolant for the leaf u, only the condition that γ_0 is a $\mathcal{L}(\Pi) \cap \mathcal{L}(\Omega)$-formula might fail. Non-axiomatic leafs are interpolated by free variables, which become bound as soon as the construction reaches the associated node of the leaf in the induction step. In the induction step, the pre-interpolant for the conclusion of a rule is built from the already constructed pre-interpolants for its premises. That is, we show how to construct a pre-interpolant for the root of t by starting at the leaves and step by step working through the proof tree. We then show that the constructed pre-interpolant is indeed an interpolant for $\Gamma \Rightarrow \Delta$. Throughout the proof of the Craig interpolation theorem we write $\vdash \Pi \Rightarrow \Omega$ for $\mathbf{C2DT} \vdash \Pi \Rightarrow \Omega$.

Proof. (of theorem 6.3.2) Suppose that $\Gamma \Rightarrow \Delta$ is valid for $\Gamma, \Delta \subseteq \Sigma_1^\mu$ and $L(\Gamma) \cap L(\Delta) \neq \emptyset$. By theorem 6.2.9 there exists a circular proof $t = (V, \rightarrow, \lambda)$ for $\Gamma \Rightarrow \Delta$. We construct an interpolant for $\Gamma \Rightarrow \Delta$ by induction on t.

Base case: We show how to obtain pre-interpolants for the leafs of t. We distinguish two cases:

- ▷ Suppose $u \in V$ is an axiomatic leaf. Then we distinguish four cases:

  1. Suppose u is labelled by $\Pi, P, \neg P \Rightarrow \Omega$. Let $Q \in At(\Gamma) \cap At(\Delta)$ (recall that this set is non-empty by assumption) and define $\gamma_0 := Q \wedge \neg Q$. First of all notice that γ_0 is a $L(\Gamma) \cap L(\Delta)$-formula. Moreover, $\vdash \Pi, P, \neg P \Rightarrow Q \wedge \neg Q$ as this is an instance of $(Ax)_4$ and $\vdash Q \wedge \neg Q \Rightarrow \Omega$ as the following derivation shows:
  $$\dfrac{\dfrac{}{Q, \neg Q \Rightarrow \Omega}\,(Ax)_4}{Q \wedge \neg Q \Rightarrow \Omega}\,(\wedge)_L$$
  2. Suppose u is labelled by $\Pi \Rightarrow P, \neg P, \Omega$. Let $Q \in At(\Gamma) \cap At(\Delta)$ and define $\gamma_0 := Q \vee \neg Q$. Clearly, γ_0 is a $L(\Gamma) \cap L(\Delta)$-formula and $\vdash \Pi \Rightarrow Q \vee \neg Q$ as the following derivation shows:
  $$\dfrac{\dfrac{}{\Pi \Rightarrow Q, \neg Q}\,(Ax)_2}{\Pi \Rightarrow Q \vee \neg Q}\,(\vee)_R$$
  Finally, $\vdash Q \vee \neg Q \Rightarrow P, \neg P, \Omega$ as this is an instance of $(Ax)_2$.
  3. Suppose u is labelled by $\Pi, P \Rightarrow P, \Omega$. Notice that this implies that $P \in At(\Gamma) \cap At(\Delta)$, as each formula labelling some node of t is a subformula of some formula labelling the root of t. Hence let $\gamma_0 := P$. Again γ_0 is a $L(\Gamma) \cap L(\Delta)$-formula and $\vdash \Pi, P \Rightarrow P$ and $\vdash P \Rightarrow P, \Omega$ as these are both instances of $(Ax)_1$.
  4. Suppose u is labelled by $\Pi, \neg P \Rightarrow \neg P, \Omega$. Then using the same argument as in the previous case we define the pre-interpolant to be $\gamma_0 := \neg P$.

- ▷ Suppose $u \in V$ is a non-axiomatic leaf with associated node u'. In case there exists another non-axiomatic leaf $v \in V$ which has the same associated node u' and we have already constructed a pre-interpolant[1] γ_0 for v, we define the pre-interpolant for u to be γ_0 as well. Otherwise, let X be a fresh variable and define $\gamma_0 := X$.[2]

[1] We assume without loss of generality that if u and v are non-axiomatic leafs which have the same associated node, then they also have the same distinguished formula.

[2] The variable X is fresh if and only if $X \notin Sub(\Gamma) \cup Sub(\Delta)$.

Induction step: Let v be a node in t and assume that we have already constructed pre-interpolants for v's children. We show how to define a pre-interpolant γ for v. For that we distinguish the following cases:

1. Suppose v is labelled by $\Pi, \varphi_0 \wedge \varphi_1 \Rightarrow \Omega$ and the last rule applied was $(\wedge)_L$:

$$\frac{\Pi, \varphi_0, \varphi_1 \Rightarrow \Omega}{\Pi, \varphi_0 \wedge \varphi_1 \Rightarrow \Omega}\ (\wedge)_L$$

   By induction hypothesis we have a pre-interpolant γ_0 for the child of v. In case v is not an associated node, let $\gamma := \gamma_0$. Otherwise, v is the associated node of some non-axiomatic leaf u which has pre-interpolant X. Notice that every non-axiomatic leaf has its distinguished formula on the left, as we are only considering Σ_1^μ-formulas. Let $\gamma := \mu X.\gamma_0$.

2. Suppose v is labelled by $\Pi \Rightarrow \varphi_0 \wedge \varphi_1, \Omega$ and the last rule applied was $(\wedge)_R$:

$$\frac{\Pi \Rightarrow \varphi_0, \Omega \qquad \Pi \Rightarrow \varphi_1, \Omega}{\Pi \Rightarrow \varphi_0 \wedge \varphi_1, \Omega}\ (\wedge)_R$$

   By induction hypothesis we have a pre-interpolant γ_0 for the left child of v and a pre-interpolant γ_1 for the right child of v. In case v is not an associated node, let $\gamma := \gamma_0 \wedge \gamma_1$. Otherwise let $\gamma := \mu X.(\gamma_0 \wedge \gamma_1)$ where X is the pre-interpolant of the leaf to which v is associated.

3. Suppose v is labelled by $\Pi, \varphi_0 \vee \varphi_1 \Rightarrow \Omega$ and the last rule applied is $(\vee)_L$:

$$\frac{\Pi, \varphi_0 \Rightarrow \Omega \qquad \Pi, \varphi_1 \Rightarrow \Omega}{\Pi, \varphi_0 \vee \varphi_1 \Rightarrow \Omega}\ (\vee)_L$$

   By induction hypothesis we have pre-interpolants γ_0 and γ_1 for the left and right child of v respectively. In case v is not an associated node, let $\gamma := \gamma_0 \vee \gamma_1$, otherwise let $\gamma := \mu X.(\gamma_0 \vee \gamma_1)$ where X is the pre-interpolant of the leaf to which v is associated.

4. The cases where the last rule applied $(\vee)_R, (Z)_L, (Z)_R, (\mu)_L$ or $(\mu)_R$ are all identical to the first case. That is, if v is not an associated node, then we let the pre-interpolant for v be the same as for its child. Otherwise, we bind the pre-interpolant of the child by a μ-operator.

5. Suppose v is labelled by $\Sigma, \Box\Pi, \Diamond\varphi \Rightarrow \Diamond\Omega, \Theta$ and the last rule applied is $(\Box)_L$:

$$\frac{\Pi, \varphi \Rightarrow \Omega}{\Sigma, \Box\Pi, \Diamond\varphi \Rightarrow \Diamond\Omega, \Theta}\ (\Box)_L$$

By induction hypothesis we have a pre-interpolant γ_0 for the child of v. In case v is not an associated node, let $\gamma := \Diamond\gamma_0$. Otherwise, let $\gamma := \mu X.\Diamond\gamma_0$ where X is the pre-interpolant of the leaf to which v is associated.

6. Suppose v is labelled by $\Sigma, \Box\Pi \Rightarrow \Box\varphi, \Diamond\Omega, \Theta$ and the last rule applied is $(\Box)_R$:

$$\frac{\Pi \Rightarrow \varphi, \Omega}{\Sigma, \Box\Pi \Rightarrow \Box\varphi, \Diamond\Omega, \Theta}\ (\Box)_R$$

By induction hypothesis we have a pre-interpolant γ_0 for the child of v. In case v is not an associated node, let $\gamma := \Box\gamma_0$. Otherwise, let $\gamma := \mu X.\Box\gamma_0$ where X is the pre-interpolant of the leaf to which v is associated.

Following this construction we obtain a pre-interpolant γ for the sequent $\Gamma \Rightarrow \Delta$ labelling the root. We show that γ is indeed an interpolant for $\Gamma \Rightarrow \Delta$.

First of all, notice that all pre-interpolants for the leafs of t are formulas in the common language of Γ and Δ. Therefore, the pre-interpolant γ for the root is by construction a $L(\Gamma) \cap L(\Delta)$-formula. We show that $\Gamma \Rightarrow \gamma$ is valid by constructing a circular proof for $\Gamma \Rightarrow \gamma$ using $t = (V, \to, \lambda)$. Given a circular pre-proof $t' = (V', \to', \lambda')$ for $\Gamma \Rightarrow \gamma$ and nodes $u \in V$ and $u' \in V'$, we say that u and u' are *corresponding* - written $u \leftrightsquigarrow u'$ - if u is labelled by $\Pi \Rightarrow \Omega$ and u' by $\Pi \Rightarrow \Theta$, such that the pre-interpolant γ_u which is assigned to u belongs to Θ. Therefore, two nodes u and u' are corresponding if they are labelled by the same formulas on the left and the pre-interpolant of u labels the right side of u'. Recall that paths in circular proofs are allowed to pass through non-axiomatic leafs and therefore be infinite. For this proof we assume that paths are finite and end in axiomatic or non-axiomatic leafs. Given two such finite paths $\mathbb{P} = \mathbb{P}(0)...\mathbb{P}(n)$ through t and $\mathbb{P}' = \mathbb{P}'(0)...\mathbb{P}'(m)$ through t', we say that $\mathbb{P}$ and $\mathbb{P}'$ are *corresponding* - written $\mathbb{P} \leftrightsquigarrow \mathbb{P}'$ - if for every $0 \le i \le n$ there exists $j \le m$ such that $\mathbb{P}(i) \leftrightsquigarrow \mathbb{P}'(j)$ and if $i \le i'$, then given that j and j' index the corresponding nodes of $\mathbb{P}(i)$ and $\mathbb{P}(i')$, we have that $j \le j'$. Notice that we allow the case that several nodes in $\mathbb{P}$ correspond to the same node in $\mathbb{P}'$ and also that there are some nodes in $\mathbb{P}'$ which do not correspond to any nodes in $\mathbb{P}$. We now show how to construct $t' = (V', \to', \lambda')$:

- ▷ The root $r_{t'}$ of t' is labelled by $\Gamma \Rightarrow \gamma$. Notice that $r_t \leftrightsquigarrow r_{t'}$.
- ▷ Suppose we have constructed $\mathbb{P}'(0)...\mathbb{P}'(m)$ where $\mathbb{P}'(0) = r_{t'}$ which corresponds to the initial segment $\mathbb{P}(0)...\mathbb{P}(n)$ of a path $\mathbb{P}$ through t, where $\mathbb{P}(n)$ is not a leaf. Moreover we assume that $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(m)$. We show how to extend $\mathbb{P}'$:
  - Suppose $\mathbb{P}(n)$ is labelled by $\Pi, \varphi_0 \wedge \varphi_1 \Rightarrow \Omega$ and the rule applied is $(\wedge)_L$ to generate the node $\mathbb{P}(n+1)$ which is labelled by $\Pi, \varphi_0, \varphi_1 \Rightarrow \Omega$. Since $\mathbb{P}(n) \leftrightsquigarrow \mathbb{P}'(m)$ we have that $\mathbb{P}'(m)$ is labelled by $\Pi, \varphi_0 \wedge \varphi_1 \Rightarrow \gamma_n, \Theta$ where γ_n is the pre-interpolant

assigned to $\mathbb{P}(n)$. If $\mathbb{P}(n)$ is not an associated node, apply the rule $(\wedge)_L$ to generate $\mathbb{P}'(m+1)$ which is labelled by $\Pi, \varphi_0, \varphi_1 \Rightarrow \gamma_n, \Theta$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+1)$ as by construction the pre-interpolant of $\mathbb{P}(n+1)$ is γ_n. If $\mathbb{P}(n)$ is an associated node to the leaf u, then $\gamma_n = \mu X.\gamma_{n+1}$ where X is the pre-interpolant of u and γ_{n+1} is the pre-interpolant of $\mathbb{P}(n+1)$. Apply the rule $(\mu)_R$, then the rule $(X)_R$ to decompose and regenerate the body of the fixed point formula γ_{n+1} and then the rule $(\wedge)_L$ to generate $\mathbb{P}'(m+3)$ labelled by $\Gamma, \varphi_0, \varphi_1 \Rightarrow \gamma_{n+1}, \Theta$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+3)$ and so also $\mathbb{P}(0)...\mathbb{P}(n)\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(0)...\mathbb{P}'(m)...\mathbb{P}'(m+3)$. In the following paragraphs we do no longer mention that the paths are corresponding.

- Suppose $\mathbb{P}(n)$ is labelled by $\Pi \Rightarrow \varphi_0 \wedge \varphi_1, \Omega$ and the rule applied is $(\wedge)_R$ which generates two nodes $\mathbb{P}(n+1)$ and v where without loss of generality $\mathbb{P}(n+1)$ is labelled by $\Pi \Rightarrow \varphi_0, \Omega$ and v is labelled by $\Pi \Rightarrow \varphi_1, \Omega$. First assume that $\mathbb{P}(n)$ is not an associated node. By assumption $\mathbb{P}'(m)$ is labelled by $\Pi \Rightarrow \gamma_n, \Theta$. Notice that by construction γ_n is of the form $\gamma_{n+1} \wedge \gamma_v$ where γ_{n+1} is the pre-interpolant of $\mathbb{P}(n+1)$ and γ_v is the pre-interpolant of v. Thus apply the rule $(\wedge)_R$ to generate two nodes u' and v' labelled by $\Pi \Rightarrow \gamma_{n+1}, \Theta$ and $\Pi \Rightarrow \gamma_v, \Theta$ respectively. Let $\mathbb{P}'(m+1) = u'$ and notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+1)$. Second, assume that $\mathbb{P}(n)$ is an associated node to the non-axiomatic leaf u. Then γ_n is of the form $\mu X.\gamma_{n+1} \wedge \gamma_v$. Thus first apply the rule $(\mu)_R$, then $(X)_R$ and then $(\wedge)_R$ to generate two new nodes u' and v' labelled as before and let $\mathbb{P}'(m+3) = u'$. Again we have that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+3)$.
- Suppose $\mathbb{P}(n)$ is labelled by $\Pi, \varphi_0 \vee \varphi_1 \Rightarrow \Omega$ and the rule applied is $(\vee)_L$ which generates two nodes $\mathbb{P}(n+1)$ and v which are labelled without loss of generality by $\Pi, \varphi_0 \Rightarrow \Omega$ and $\Pi, \varphi_1 \Rightarrow \Omega$ respectively. By assumption $\mathbb{P}'(m)$ is labelled by $\Pi, \varphi_0 \vee \varphi_1 \Rightarrow \gamma_n, \Theta$. First suppose $\mathbb{P}(n)$ is not an associated node. Then γ_n is by construction of the form $\gamma_{n+1} \vee \gamma_v$. Thus first apply the rule $(\vee)_R$ to generate the node $\mathbb{P}'(m+1)$ labelled by $\Pi, \varphi_0 \vee \varphi_1 \Rightarrow \gamma_{n+1}, \gamma_v, \Theta$ and then apply the rule $(\vee)_L$ to generate two new nodes u' and v' labelled by $\Pi, \varphi_0 \Rightarrow \gamma_{n+1}, \gamma_v, \Theta$ and $\Pi, \varphi_1 \Rightarrow \gamma_{n+1}, \gamma_v, \Theta$ respectively. Let $\mathbb{P}'(m+2) = u'$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+2)$. Second suppose that $\mathbb{P}(n)$ is an associated node. Then γ_n is of the form $\mu X.\gamma_{n+1} \vee \gamma_v$ and we first apply the rule $(\mu)_R$, then $(X)_R$, then $(\vee)_R$ and finally $(\vee)_L$ which generates two nodes u' and v' labelled as before. Then let $\mathbb{P}'(m+4) = u'$ and notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+4)$.
- Suppose $\mathbb{P}(n)$ is labelled by $\Pi \Rightarrow \varphi_0 \vee \varphi_1, \Omega$ and the rule applied is $(\vee)_R$ to generate the node $\mathbb{P}(n+1)$ labelled by $\Pi \Rightarrow \varphi_0, \varphi_1, \Omega$. By assumption $\mathbb{P}'(m)$ is labelled by $\Pi \Rightarrow \gamma_n, \Theta$. If $\mathbb{P}(n)$ is not an associated node, then $\gamma_n = \gamma_{n+1}$ by definition and so we do not apply any rule and observe that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m)$. Otherwise, $\gamma_n = \mu X.\gamma_{n+1}$ and we apply the rules $(\mu)_R$ and then $(X)_R$ to generate $\mathbb{P}'(m+2)$ and notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+2)$.
- Suppose $\mathbb{P}(n)$ is labelled by $\Sigma, \Box\Pi, \Diamond\varphi \Rightarrow \Diamond\Omega, \Psi$ and the rule applied is $(\Box)_L$ to generate the node $\mathbb{P}(n+1)$ labelled by $\Pi, \varphi \Rightarrow \Omega$. By assumption $\mathbb{P}'(m)$ is labelled

by $\Sigma, \Box\Pi, \Diamond\varphi \Rightarrow \gamma_n, \Theta$. If $\mathbb{P}(n)$ is not an associated node, then $\gamma_n = \Diamond\gamma_{n+1}$. Therefore apply the rule $(\Box)_L$ to generate the node $\mathbb{P}'(m+1)$ which is labelled by $\Pi, \varphi \Rightarrow \gamma_{n+1}$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+1)$. Otherwise γ_n is of the form $\mu X.\Diamond\gamma_{n+1}$ and we first apply $(\mu)_R$, $(X)_R$ and then $(\Box)_L$ to generate $\mathbb{P}'(m+3)$ labelled by $\Pi, \varphi \Rightarrow \gamma_{n+1}$ and $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+3)$.

- Suppose $\mathbb{P}(n)$ is labelled by $\Sigma, \Box\Pi \Rightarrow \Box\varphi, \Diamond\Omega, \Psi$ and the rule applied is $(\Box)_R$ to generate the node $\mathbb{P}(n+1)$ labelled by $\Pi \Rightarrow \varphi, \Omega$. By assumption $\mathbb{P}'(m)$ is labelled by $\Sigma, \Box\Pi \Rightarrow \gamma_n, \Theta$. If $\mathbb{P}(n)$ is not an associated node, then $\gamma_n = \Box\gamma_{n+1}$. Apply the rule $(\Box)_R$ to generate the node $\mathbb{P}'(m+1)$ labelled by $\Pi \Rightarrow \gamma_{n+1}$ and observe that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+1)$. Otherwise $\gamma_n = \mu X.\Box\gamma_{n+1}$ and we first apply $(\mu)_R$, $(X)_R$ and then $(\Box)_R$ to generate $\mathbb{P}'(m+3)$, where $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+3)$.
- Suppose $\mathbb{P}(n)$ is labelled by $\Pi, Z \Rightarrow \Omega$ and the rule applied is $(Z)_L$ to generate $\mathbb{P}(n+1)$ labelled by $\Pi, \varphi(Z) \Rightarrow \Omega$. By assumption $\mathbb{P}'(m)$ is labelled by $\Pi, Z \Rightarrow \gamma_n, \Theta$. If $\mathbb{P}(n)$ is not an associated node, then $\gamma_n = \gamma_{n+1}$ and so we apply the rule $(Z)_L$ to generate the node $\mathbb{P}'(m+1)$ labelled by $\Pi, \varphi(Z) \Rightarrow \gamma_n, \Theta$ and notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+1)$. Otherwise $\gamma_n = \mu X.\gamma_{n+1}$ and we apply first $(\mu)_R$, $(X)_R$ and then $(Z)_L$ to generate $\mathbb{P}'(m+3)$ labelled by $\Pi, \varphi(Z) \Rightarrow \gamma_{n+1}, \Theta$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+3)$.
- Suppose $\mathbb{P}(n)$ is labelled by $\Pi \Rightarrow Z, \Omega$ and the rule applied is $(Z)_R$ to generate the node $\mathbb{P}(n+1)$ labelled by $\Pi \Rightarrow \varphi(Z), \Omega$. By assumption $\mathbb{P}'(m)$ is labelled by $\Pi \Rightarrow \gamma_n, \Theta$. If $\mathbb{P}(n)$ is not an associated node, then $\gamma_n = \gamma_{n+1}$. Therefore we do not apply any rule as $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m)$. Otherwise $\gamma_n = \mu X.\gamma_{n+1}$ and we apply $(\mu)_R$ and $(X)_R$ to generate $\mathbb{P}'(m+2)$ labelled by $\Pi \Rightarrow \gamma_{n+1}, \Theta$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+2)$.
- Suppose $\mathbb{P}(n)$ is labelled by $\Pi, \mu Z.\varphi(Z) \Rightarrow \Omega$ and the rule applied is $(\mu)_L$ to generate the node $\mathbb{P}(n+1)$ labelled by $\Pi, Z \Rightarrow \Omega$. By assumption $\mathbb{P}'(m)$ is labelled by $\Pi, \mu Z.\varphi(Z) \Rightarrow \gamma_n, \Theta$. If $\mathbb{P}(n)$ is not an associated node, then $\gamma_n = \gamma_{n+1}$. Thus apply the rule $(\mu)_L$ to generate the node $\mathbb{P}'(m+1)$ labelled by $\Pi, Z \Rightarrow \gamma_{n+1}, \Theta$ and notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+1)$. Otherwise $\gamma_n = \mu X.\gamma_{n+1}$ and we apply $(\mu)_R$, then $(X)_R$ and then $(\mu)_L$ to generate $\mathbb{P}'(m+3)$ labelled by $\Pi, Z \Rightarrow \gamma_{n+1}, \Theta$. Observe that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+3)$.
- Suppose $\mathbb{P}(n)$ is labelled by $\Pi \Rightarrow \mu Z.\varphi(Z), \Omega$ and the rule applied is $(\mu)_R$ to generate the node $\mathbb{P}(n+1)$ labelled by $\Pi \Rightarrow Z, \Omega$. By assumption $\mathbb{P}'(m)$ is labelled by $\Pi \Rightarrow \gamma_n, \Theta$. If $\mathbb{P}(n)$ is not an associated node, then $\gamma_n = \gamma_{n+1}$ and we do not apply any rule as $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m)$. Otherwise, $\gamma_n = \mu X.\gamma_{n+1}$ and we apply $(\mu)_R$ and $(X)_R$ to generate $\mathbb{P}'(m+2)$ labelled by $\Pi \Rightarrow \gamma_{n+1}, \Theta$. Notice that $\mathbb{P}(n+1) \leftrightsquigarrow \mathbb{P}'(m+2)$.

When the construction reaches the leaves of the circular proof t, we have built a finite tree t'_0 whose root is labelled by $\Gamma \Rightarrow \gamma$ and which is built by the rules of **C2DT**. Furthermore it has the property that every path through t'_0 corresponds to a path through t. However, t'_0 is not yet a circular proof. Observe that leafs of t'_0 which correspond to non-axiomatic leafs in t are not (in general) non-axiomatic themselves. This is because the first node after the associated node might be labelled by a different side sequent on the right than the leaf. Moreover, leafs corresponding to axiomatic leafs labelled by $(Ax)_2$ are not yet axiomatic leafs either. We show how to finish the construction and simultaneously prove that the constructed tree is a circular proof. We do this in two steps:

1. Suppose u' is a leaf of t'_0 which corresponds to an axiomatic leaf u of t. If u is labelled by $(Ax)_1, (Ax)_3$ or $(Ax)_4$, then u' is labelled by $(Ax)_1, (Ax)_3$ or $(Ax)_4$ as well. So in all three cases u' is an axiomatic leaf and we do not extend the tree. If u is labelled by $(Ax)_2$, namely $\Pi \Rightarrow P, \neg P, \Omega$, then u' is labelled by $\Pi \Rightarrow Q \vee \neg Q, \Theta$. In that case we extend the tree by applying the rule $(\vee)_R$ to the formula $Q \vee \neg Q$ to generate u'' which is labelled by $\Pi \Rightarrow Q, \neg Q, \Theta$ and which is therefore an axiomatic leaf.

2. Suppose u' is a leaf of t'_0 which corresponds to a non-axiomatic leaf u of t. Let v be the associated node of u and suppose that u is labelled by $\Pi, \varphi \Rightarrow \Omega$ where φ is the distinguished formula. Since u and u' are corresponding we have that u' is labelled by $\Pi, \varphi \Rightarrow X, \Theta$ where the corresponding node of v, say v', is labelled by $\Pi, \varphi \Rightarrow \gamma_v, \Theta'$ where $\gamma_v = \mu X.\gamma_{v+1}$ and $v+1$ is the successor of v. Notice that the successor node of v' is labelled by $\Pi, \varphi \Rightarrow X, \Theta'$. This means that u' is *almost* a non-axiomatic leaf with the successor of v' as its associated node. The only shortcoming is that the side sequents Θ and Θ' might be different. However, every rule applied between the successor of v' and u' is applied either to a formula on the left or to a pre-interpolant on the right. This means that the side sequent Θ' is irrelevant for building the steps between these two nodes. We assume that there are n steps between the successor of v' and u'. Therefore, we can continue to extend the branch that leads to u' by applying the same rules in the same order to the same formulas as between the successor of v' and u'. In doing so, we find a repetition, say at node u'' with associated node v'', in at most n steps. Notice that the trace which passes through φ both at node v' and u' is extended up to the node u''. Therefore the formula labelling u'' on the left which belongs to that trace is the distinguished formula of u''. Notice that this formula belongs to a trace that starts in a Σ_1^μ-formula on the left. Hence, u'' is a non-axiomatic leaf and we extend the branch up to u''.

Therefore we can extend t'_0 into the finite tree t', such that each leaf of t' is either axiomatic or non-axiomatic. This implies that t' is a circular proof for $\Gamma \Rightarrow \gamma$ and therefore, by soundness of **C2DT**, that $\Gamma \Rightarrow \gamma$ is valid.

In order to show that $\gamma \Rightarrow \Omega$ is valid, we build a prooftree in the same way as above, where two nodes u and u' are now called corresponding if and only if u is labelled by $\Pi \Rightarrow \Omega$ and u' is labelled by $\Theta, \gamma_u \Rightarrow \Omega$. Each step of the construction is symmetric to the construction above. Notice that for non-axiomatic leafs, as the distinguished formula always occurs on the left, the constructed pre-interpolant serves as the distinguished formula. This gives us valid non-axiomatic leafs, since we constructed the pre-interpolant using μ-operators. We do not provide details of that direction and simply conclude that $\vdash \gamma \Rightarrow \Omega$ and therefore that $\gamma \Rightarrow \Omega$ is valid. Together we conclude that γ is indeed a Craig interpolant for $\Gamma \Rightarrow \Delta$. $\square$

Observe that the constructed interpolant belongs itself to the fragment Σ_1^μ. That is, we find optimal interpolants belonging to the the same fragment as the formulas in the sequent.

Establishing Craig interpolation for Π_1^μ is essentially the same argument as for Σ_1^μ. The only step that is changed is the binding of free variable in associated nodes. Previously, we bound the variable by a μ-operator, as every distinguished formula of a non-axiomatic leaf occurs on the left. In circular proofs for Π_1^μ-sequents it is the other way around. The distinguished formula always occurs on the right and so we bind the variable by a ν-operator. Apart from this small change, the proof is completely symmetric. We therefore omit the details and simply state the theorem.

Theorem 6.3.3 (Craig interpolation for Π_1^μ)**.** Let $\Gamma, \Delta \subseteq \Pi_1^\mu$. If $\Gamma \Rightarrow \Delta$ is valid and $\mathcal{L}(\Gamma) \cap \mathcal{L}(\Delta) \neq \emptyset$, then there exists $\gamma \in \mathcal{L}(\Gamma) \cap \mathcal{L}(\Delta)$ such that $\Gamma \Rightarrow \gamma$ and $\gamma \Rightarrow \Delta$ are valid.

Observe that the constructed interpolant for Π_1^μ-sequents belongs to Π_1^μ, which implies that the construction is once again optimal.

6.4 Craig interpolation for $\Sigma_1^\mu \cup \Pi_1^\mu$

In this section we combine theorem 6.3.2 and theorem 6.3.3 into a result that the first level of the alternation hierarchy $\Sigma_1^\mu \cup \Pi_1^\mu$ enjoys Craig interpolation. There is one problem: If sequents contain formulas from both Σ_1^μ and Π_1^μ, then circular proofs for such sequents have non-axiomatic leafs where the distinguished formula occurs on the left and others where it occurs on the right. Whenever the distinguished formula occurs on the left, we bind the pre-interpolant by a μ-operator and whenever it occurs on the right, by a ν-operator. Therefore, we create an interpolant γ which in general does not belong to the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$, as it might contain fixed point alternation. The alternation is introduced into the interpolant when two non-axiomatic leafs with distinguished formulas on opposite sides have their associated nodes in the same branch. We show in the next section, that the interpolant can be optimized to a certain degree. Namely, we find interpolants that belong to the alternation-free fragment $\Sigma_2^\mu \cap \Pi_2^\mu$, by analysing the structure in which non-axiomatic leafs occur in a circular proof. For now, we ignore this issue and construct interpolants with fixed point alternation. Another problem that arises from having fixed point alternation in the interpolant, is that we can no longer use the circular sequent calculus **C2DT** to show that $\Gamma \Rightarrow \gamma$ and $\gamma \Rightarrow \Delta$ is valid, as **C2DT** is only sound with respect to $\Sigma_1^\mu \cup \Pi_1^\mu$. We solve this by turning towards the system **2DT** for proving that $\Gamma \Rightarrow \gamma$ and $\gamma \Rightarrow \Delta$ are valid (while we still use the circular calculus to construct the interpolant).

Theorem 6.4.1 (Craig interpolation for $\Sigma_1^\mu \cup \Pi_1^\mu$)**.** Let $\Gamma, \Delta \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$. If $\Gamma \Rightarrow \Delta$ is valid and $\mathcal{L}(\Gamma) \cap \mathcal{L}(\Delta) \neq \emptyset$, then there exists a $\mathcal{L}(\Gamma) \cap \mathcal{L}(\Delta)$-formula γ such that $\Gamma \Rightarrow \gamma$ is valid and $\gamma \Rightarrow \Delta$ is valid.

Proof. Suppose that $\Gamma \Rightarrow \Delta$ is valid. This implies that there is a **C2DT**-proof $t = (V, \to, \lambda)$ of $\Gamma \Rightarrow \Delta$. We construct an interpolant for $\Gamma \Rightarrow \Delta$ by induction on t.

Base case: The base case is identical to the base case of the construction of the interpolant in theorem 6.3.2.

Induction step: Let v be a node in t and suppose we have already constructed pre-interpolants for v's children. We then construct a pre-interpolant for v in the same way as in the induction step of the construction of the interpolant in theorem 6.3.2 and theorem 6.3.3 with the following difference: If v is an associated node to the non-axiomatic leaf u, then we bind the pre-interpolant for v by a μ-operator just in case the distinguished formula of u occurs on the left and by a ν-operator otherwise.

Let γ be the pre-interpolant for $\Gamma \Rightarrow \Delta$ which is constructed in the above described way. We show that γ is an interpolant. First of all, observe that γ belongs to the common language of Γ and Δ by construction. We show that $\Gamma \Rightarrow \gamma$ is valid, by proving that $\Gamma \Rightarrow \gamma$ is **2DT**-derivable. Recall that t is the circular proof for $\Gamma \Rightarrow \Delta$. We build the finite tree $t_0' = (V_0', \to_0', \lambda_0')$ in the same way as in the proof of theorem 6.3.2. So t_0' has the following properties:

1. The root of t_0' is labelled by $\Gamma \Rightarrow \gamma$.
2. t_0' is generated by the rules from **C2DT** (which are the rules of **2DT**).
3. Every leaf of t_0' either corresponds to an axiomatic leaf of t or it corresponds to a non-axiomatic leaf of t.

As we observed in the proof of Craig interpolation for Σ_1^μ, the tree t_0' is almost a circular proof for $\Gamma \Rightarrow \gamma$. The only shortcomings of t_0' were those leafs which correspond to non-axiomatic leafs of t, as they were not yet non-axiomatic themselves, as well as those corresponding to axiomatic leafs labelled by $(Ax)_2$. However, we also observed that this is not an issue, as we can extend the relevant branches to turn t_0' into a circular proof. Notice that each leaf of t_0' which does not correspond to an axiomatic leaf is labelled by some fresh variable X on the right side, where X is the pre-interpolant of the corresponding non-axiomatic leaf of t. In the following paragraph we call those leafs of t_0' which correspond to non-axiomatic leafs of t non-axiomatic as well. Moreover, given a non-axiomatic leaf u of t_0' corresponding to v, we call the node that corresponds to the associated node of v the associated node of u. We show how to extend t_0' into a suitable prooftree.

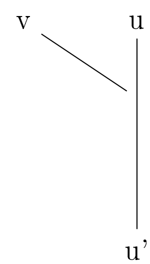
First of all, if the constructed pre-interpolant γ belongs to $\Sigma_1^\mu \cup \Pi_1^\mu$, then t_0' can be extended into a circular proof t' for $\Gamma \Rightarrow \gamma$ in the same way as in the proof of theorem 6.3.2. Therefore suppose γ does not belong to $\Sigma_1^\mu \cup \Pi_1^\mu$. We unravel t_0' over its non-axiomatic leafs and their corresponding nodes into an infinitary **2DT**-pre-proof $\hat{t}$ as described in the soundness proof of **C2DT**. Notice that every leaf of $\hat{t}$ corresponds to an axiomatic leaf of t, which implies that every leaf of $\hat{t}$ is itself axiomatic (or can be turned into an axiomatic leaf in one step). Therefore we only have to show that every infinite path through $\hat{t}$ contains an infinite trace of the right form. So suppose $\mathbb{P}$ is an infinite path through $\hat{t}$. Since $\hat{t}$ is the unravelling of t_0', the path $\mathbb{P}$ passes through some non-axiomatic leaf of t_0' infinitely often. Now given that t_0' is a finite tree and has therefore only finitely many non-axiomatic leafs, $\mathbb{P}$ passes through at most finitely many non-axiomatic leafs $v_1, ..., v_k$ infinitely often. Recall that each of these leafs is labelled by some variable $X_1, ..., X_k$ on the right, where without loss of generality $X_i \neq X_j$ for each $1 \leq i < j \leq k$. Let $u_1, ..., u_k$ be the associated nodes of $v_1, ..., v_k$. As $\mathbb{P}$ passes through all these leafs infinitely often, there exists one leaf v_i whose associated node u_i occurs below every associated node u_j of v_j for $j \neq i$. The construction of the interpolant γ therefore implies that the variable X_i subsumes the variables X_j for all $j \neq i$. From this it follows that $\mathbb{P}$ contains an infinite trace of the right form. We distinguishing two cases: First, suppose the distinguished formula φ of the non-axiomatic leaf of t which corresponds to v_i occurs on the left. This implies that v_i is labelled by $\Pi, \varphi \Rightarrow X_i, \Theta$. By definition φ belongs to a trace that starts in a Σ_1^μ-formula. Since $\mathbb{P}$ passes through v_i infinitely often there exists an infinite trace tr that passes through φ infinitely often, which implies that tr passes through a μ-variable infinitely often. Since the left side of each node is labelled by $\Sigma_1^\mu \cup \Pi_1^\mu$-formulas only, the trace tr is a μ-trace and so $\mathbb{P}$ has an infinite μ-trace on the left. Second, suppose that the distinguished formula φ of the non-axiomatic leaf of t which corresponds to v_i occurs on the right. This implies that X_i is a ν-variable. Consider the trace tr through $\mathbb{P}$ that starts on the right side with the formula γ and follows the pre-interpolants, passing through X_i infinitely often. Every variable that occurs infinitely often in tr is of the form X_j for $1 \leq j \leq k$. As X_i subsumes all these variables, we conclude that tr is an infinite ν-trace. Therefore every infinite path of $\hat{t}$ has an infinite μ-trace on the left or an infinite ν-trace on the right. This yields that $\hat{t}$ is a **2DT**-proof of $\Gamma \Rightarrow \gamma$. From soundness of **2DT** it follows that $\Gamma \Rightarrow \gamma$ is valid.

As in the proof of theorem 6.3.2, we omit the argument that $\gamma \Rightarrow \Delta$ is valid. The proof strategy for it is identical to the proof strategy above and each step in the construction is symmetric. We conclude that the constructed formula γ is an interpolant for $\Gamma \Rightarrow \Delta$. $\square$

6.5 Optimizing Craig interpolation

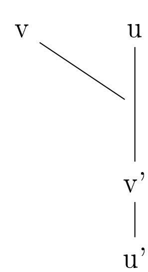
The interpolant constructed in the proof of Craig interpolation for $\Sigma_1^\mu \cup \Pi_1^\mu$ belongs to an arbitrary high level of the fixed point alternation depth hierarchy, depending on the structure of the specific circular proof which is used. In this section we discuss how to optimize the interpolant in terms of fixed point alternation.

Let us start by considering how fixed point alternation is introduced into the constructed interpolant. Suppose we are given a circular proof $t = (V, \rightarrow, \lambda)$ for some sequent $\Gamma \Rightarrow \Delta$. The introduction of fixed point alternation into the interpolant for $\Gamma \Rightarrow \Delta$ stems from non-axiomatic leafs in the proof tree, such that the associated node of one non-axiomatic leaf occurs below the associated nodes of the other non-axiomatic leafs. That is, fixed point alternation stems from situations that look as follows:



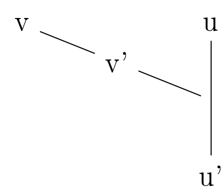
where v and u are non-axiomatic leafs and the associated node of v occurs above u'. Now suppose that the distinguished formula of u occurs on the left and the distinguished formula of v occurs on the right. When the construction of the interpolant reaches the node u', the pre-interpolant constructed so far contains a ν-operator which was introduced at the associated node of v. Since u' is the associated node of u, we have to bind the pre-interpolant by a μ-operator, which therefore introduces fixed point alternation into the interpolant. Observe that we can reduce the fixed point alternation depth in the interpolant by choosing the distinguished formulas of the non-axiomatic leafs more wisely. For example, if the leaf u is also labelled by a Π_1^μ-formula on the right, we can redefine the distinguished formula of u, such that it occurs on the right. When we build the interpolant, we then no longer introduce fixed point alternation at u'. This observation raises the question whether it is possible to get rid of any fixed point alternation by redefining the distinguished formulas in the non-axiomatic leafs? In order to answer this question, let us take a closer look at the possible positions of the associated node v' of v.

1. Suppose v' occurs in the branch between u' and u. We are therefore in the following situation:

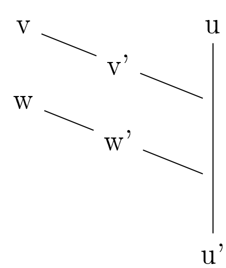


Suppose that the distinguished formula of u occurs on the left and the distinguished formula of v occurs on the right, which implies that fixed point alternation is introduced into the interpolant. However, observe that in this case the leaf u must be labelled by a Π_1^μ-formula on the right as well. If this was not the case, then u' would not be labelled by a Π_1^μ-formula on the right either, which would imply that v' and thus also v is not labelled by Π_1^μ-formulas on the right, contradicting our assumption. Therefore we let the distinguished formula of u be the said Π_1^μ-formula on the right, which implies that we do no longer introduce fixed point alternation of degree higher than 1 in this case.

2. Suppose v' does not occur on the branch between u' and u. Hence, we are in the following situation:



We can use a similar argument as in case 1. If the distinguished formula of u occurs without loss of generality on the left and the distinguished formula of v on the right, then again we find a suitable formula labelling u on the right. Unfortunately, problems arise when there are (at least) two branches branching off the path leading to u, such that both lead into non-axiomatic leafs whose associated nodes do not occur between u' and u:



Suppose the distinguished formula of u occurs on the left, the distinguished formula of v on the right and the one of w on the left again. In case v and w are labelled by formulas on both sides, we can iterate the previous argument. However, if both v and w are only labelled by formulas on one side, it is no longer possible to build the interpolant without introducing fixed point alternation, as the following example illustrates.

Example 6.5.1. We consider the following circular proof for the sequent

$$\mu X.\Diamond(((\mu Y.\Diamond Y) \vee X) \wedge P) \Rightarrow P \vee \nu Z.\Box(Z \wedge \nu V.\Box V)$$

Notice that this sequent lies in $\Sigma_1^\mu \cup \Pi_1^\mu$ and its common language is based on $\{P\}$.

$$
\begin{prooftree}
\AxiomC{$Y \Rightarrow$}
\RightLabel{$(\Box)_L$}
\UnaryInfC{$\Diamond Y \Rightarrow$}
\RightLabel{$(Y)_L$}
\UnaryInfC{$Y \Rightarrow$}
\RightLabel{$(\Box)_L$}
\UnaryInfC{$\Diamond Y, P \Rightarrow Z$}
\RightLabel{$(Y)_L$}
\UnaryInfC{$Y, P \Rightarrow Z$}
\RightLabel{$(\mu)_L$}
\UnaryInfC{$\mu Y.\Diamond Y, P \Rightarrow Z$}
\AxiomC{$((\mu Y.\Diamond Y) \vee X) \wedge P \Rightarrow Z \wedge \nu V.\Box V$}
\RightLabel{$(\Box)_L$}
\UnaryInfC{$\Diamond(((\mu Y.\Diamond Y) \vee X) \wedge P), P \Rightarrow \Box(Z \wedge \nu V.\Box V)$}
\RightLabel{$(X)_L$}
\UnaryInfC{$X, P \Rightarrow \Box(Z \wedge \nu V.\Box V)$}
\RightLabel{$(Z)_R$}
\UnaryInfC{$X, P \Rightarrow Z$}
\RightLabel{$(\vee)_L$}
\BinaryInfC{$(\mu Y.\Diamond Y) \vee X, P \Rightarrow Z$}
\AxiomC{$\Rightarrow V$}
\RightLabel{$(\Box)_R$}
\UnaryInfC{$\Rightarrow \Box V$}
\RightLabel{$(V)_R$}
\UnaryInfC{$\Rightarrow V$}
\RightLabel{$(\Box)_R$}
\UnaryInfC{$(\mu Y.\Diamond Y) \vee X, P \Rightarrow \Box V$}
\RightLabel{$(V)_R$}
\UnaryInfC{$(\mu Y.\Diamond Y) \vee X, P \Rightarrow V$}
\RightLabel{$(\nu)_R$}
\UnaryInfC{$(\mu Y.\Diamond Y) \vee X, P \Rightarrow \nu V.\Box V$}
\RightLabel{$(\wedge)_R$}
\BinaryInfC{$(\mu Y.\Diamond Y) \vee X, P \Rightarrow Z \wedge \nu V.\Box V$}
\RightLabel{$(\wedge)_L$}
\UnaryInfC{$((\mu Y.\Diamond Y) \vee X) \wedge P \Rightarrow Z \wedge \nu V.\Box V$}
\RightLabel{$(\Box)_L$}
\UnaryInfC{$\Diamond(((\mu Y.\Diamond Y) \vee X) \wedge P) \Rightarrow P, \Box(Z \wedge \nu V.\Box V)$}
\RightLabel{$(X)_L$}
\UnaryInfC{$X \Rightarrow P, \Box(Z \wedge \nu V.\Box V)$}
\RightLabel{$(\mu)_L$}
\UnaryInfC{$\mu X.\Diamond(((\mu Y.\Diamond Y) \vee X) \wedge P) \Rightarrow P, \Box(Z \wedge \nu V.\Box V)$}
\RightLabel{$(Z)_R$}
\UnaryInfC{$\mu X.\Diamond(((\mu Y.\Diamond Y) \vee X) \wedge P) \Rightarrow P, Z$}
\RightLabel{$(\nu)_R$}
\UnaryInfC{$\mu X.\Diamond(((\mu Y.\Diamond Y) \vee X) \wedge P) \Rightarrow P, \nu Z.\Box(Z \wedge \nu V.\Box V)$}
\RightLabel{$(\vee)_R$}
\UnaryInfC{$\mu X.\Diamond(((\mu Y.\Diamond Y) \vee X) \wedge P) \Rightarrow P \vee \nu Z.\Box(Z \wedge \nu V.\Box V)$}
\end{prooftree}
$$

The prooftree consists of three branches which all lead to non-axiomatic leafs. The left-most leaf has as its distinguished formula the μ-variable Y on the left. The middle leaf has both a Σ_1^μ-formula on the left and a Π_1^ν-formula on the right. Finally, the right leaf has a ν-variable as its distinguished formula on the right. Notice that this is exactly the previously described situation. In particular, both branches that branch off the middle path lead to non-axiomatic leafs that are only labelled on one side and their associated nodes occur above the associated node of the middle leaf. Now if we want to build an interpolant, we have to bind the variable interpolating the left leaf by a μ-operator and the variable interpolating the right leaf by a ν-operator. For the variable interpolating the middle leaf, we can choose whether we want to bind it by a μ- or a ν-operator. However, since the associated node of the middle leaf appears below the associated nodes of the outer leafs, the fixed point operator which we choose to bind the variable interpolating the middle leaf has both fixed point operators for the outer leafs in its scope. Thus no matter whether we choose to bind the variable interpolating the middle leaf by a μ-operator or a ν-operator, we create fixed point alternation.

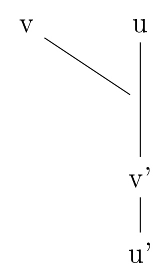
We have demonstrated that we cannot cannot construct an interpolant in the closure of $\Sigma_1^\mu \cup \Pi_1^\mu$ under Boolean connectives for every circular proof. However, observe that there exist different circular proofs for the sequent above, where the constructed interpolant does not have fixed point alternation. For example, when first applying the rule $(\Box)_L$, we could entirely delete the right side, which results in a circular proof for which the constructed interpolant only contains least fixed point operators. This raises the question whether every valid sequent has a circular proof of such a form, that the constructed interpolant belongs to the closure of $\Sigma_1^\mu \cup \Pi_1^\mu$ under Boolean connectives or whether there are sequents for which it is impossible to build such optimal interpolants? This remains an open question which we hope to answer in future research. For now, we content ourselves with the result that not every circular proof gives raise to an optimal interpolant. However, we can still establish a solid upper bound of the alternation degree for interpolants constructed with respect to arbitrary circular proofs.

Theorem 6.5.2. If $\Gamma, \Delta \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$, $\Gamma \Rightarrow \Delta$ is valid and their common language non-empty, then there exists an interpolant for $\Gamma \Rightarrow \Delta$ that lies in $\Sigma_2^\mu \cap \Pi_2^\mu$.

Proof. We first show that we find an interpolant for $\Gamma \Rightarrow \Delta$ in $\Sigma_2^\mu \cup \Pi_2^\mu$. Let $t = (V, \to, \lambda)$ be a circular proof for $\Gamma \Rightarrow \Delta$ where $\Gamma, \Delta \subseteq \Sigma_1^\mu \cup \Pi_1^\mu$ and the common language of Γ and Δ is non-empty. We consider the two cases, where fixed point alternation is introduced into the interpolant in the standard construction.

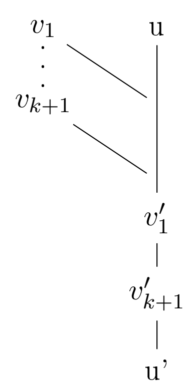
Case 1: Suppose $u \in V$ is a non-axiomatic leaf with associated node u', such that there is branching between u' and u into non-axiomatic leafs $v_1, ..., v_k$, such that their associated nodes $v_1', ..., v_k'$ occur in the branch between u' and u. We show that the distinguished formulas in all k non-axiomatic leafs can be chosen on the same side by induction on k.

- ▷ $k = 1$: There exists a non-axiomatic leaf v with associated node v' such that v' occurs in the branch between u' and u. We are therefore in the following situation:



  This case is already solved above.

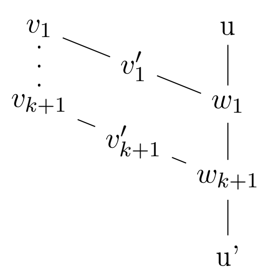
- ▷ $k \rightsquigarrow k+1$: Suppose there are $k+1$ non-axiomatic leafs $v_1, ..., v_{k+1}$ such that their associated nodes $v_1', ..., v_{k+1}'$ occur in the branch between u' and u:



Observe that in the tree above the associated nodes occur below the first branching point. This is obviously not necessarily the case, but it is depicted like this for simplicity. It does not matter for our argument in what order associated nodes and branching occurs. We assume without loss of generality that $u' \to v'_{k+1} \to v'_k \to ... \to v'_1 \to u$. By induction hypothesis the distinguished formulas of the non-axiomatic leafs $u, v_1, ..., v_k$ all occur on the same side, say without loss of generality on the left. Consider the last non-axiomatic leaf v_{k+1}. Suppose towards a contradiction that v_{k+1} is not labelled by a Σ_1^μ-formula on the left. This implies that its associated node v'_{k+1} is not labelled by a Σ_1^μ-formula on the left. Since $v'_{k+1} \to v'_k$ this implies that v'_k is not labelled by a Σ_1^μ-formula on the left and so neither is v_k, contradicting the assumption. Therefore, v_{k+1} is labelled by a Σ_1^μ-formula on the left, which we can choose to be the distinguished formula of v_{k+1}. Hence, every non-axiomatic leaf has its distinguished formula on the same side.

Therefore, in this situation we can choose the distinguished formulas in such a way that in the construction of the interpolant no fixed point alternation is introduced.

Case 2: Suppose $u \in V$ is a non-axiomatic leaf with associated node u', such that there is branching between u' and u into non-axiomatic leafs $v_1, ..., v_k$, such that their associated nodes $v'_1, ..., v'_k$ do not occur on the branch between u' and u or below u'. This is is the following situation:



If all leafs $v_1, ...v_k$ are labelled by formulas on both sides, we can use a similar argument as in case 1 to show that we can choose every distinguished formula to occur on the same side and we therefore do not introduce fixed point alternation. Fixed point alternation is only introduced when there are at least two nodes v_i and v_j which are labelled only on one side, such that v_i is labelled without loss of generality on the left and v_j on the right. Let $\sigma X_i.\gamma_i$ be the pre-interpolant for the associated node v'_i of such a non-axiomatic leaf v_i. Observe that $\sigma X_i.\gamma_i$ has syntactic alternation depth 1, as v_i is only labelled on one side. Observe that the associated nodes of v_i and v_j do not occur in the same branch. Thus, when the construction of the interpolant has reached the node u', it holds that $\sigma X_i.\gamma_i \notin Sub(\sigma X_j.\gamma_j)$ and vice versa. Therefore, when we bind the pre-interpolant at node u' by a fixed point operator, we introduce at most syntactic fixed point alternation of depth 2.

Now suppose we have a mix of these two situations. That is, there exists a non-axiomatic leaf u with associated node u' such that between u' and u there is both

- ▷ branching into non-axiomatic leafs $v_1, ..., v_k$ such that their associated nodes $v'_1, ..., v'_k$ occur in the branch between u' and u and
- ▷ branching into non-axiomatic leafs $w_1, ..., w_l$ such that their associated nodes $w'_1, ..., w'_l$ do not occur in the branch between u' and u or below u'.

By case 1 we can first choose the distinguished formulas of u and $v_1, ..., v_k$ in such a way that they all occur on the same side. Let us assume they occur without loss of generality on the left. Moreover, for every leaf of $w_1, ..., w_l$ which is labelled on both sides we can also choose the distinguished formula on the left side. Then when we build the interpolant, by case 2 every greatest fixed point operator occurs in the scope of a finite string of least fixed point operators, which implies that we introduce syntactic fixed point alternation of at most depth 2. Therefore the interpolant belongs to the fragment $\Sigma_2^\mu \cup \Pi_2^\mu$.

Let us now show that the constructed interpolant indeed lies in $\Sigma_2^\mu \cap \Pi_2^\mu$.[3] Let γ be the interpolant for $\Gamma \Rightarrow \Delta$ and let $\gamma_0 \in Sub(\gamma)$ be a subformula of γ that contains syntactic fixed point alternation of depth 2. Therefore γ_0 stems from a subtree of t of the form as in case 2. As we observed, the subformulas of γ_0 which correspond to the non-axiomatic leafs which are labelled on one side only, are of the form $\sigma Z.\varphi(Z)$ where $\sigma \in \{\mu, \nu\}$. Moreover, $\sigma Z.\varphi(Z)$ contains no syntactic fixed point alternation. Let γ'_0 be the formula γ_0 where each such subformula is replaced by a fresh variable Y. Notice that $\gamma'_0 \in \Sigma_1^\mu$ or $\gamma'_0 \in \Pi_1^\mu$. Therefore $\gamma'_0 \in \Sigma_2^\mu \cap \Pi_2^\mu$ and contains $s \geq 2$ free variables. Moreover each subformula of the form $\sigma Z.\varphi(Z)$ which is replaced by a variable in γ'_0 belongs to $\Sigma_1^\mu \cup \Pi_1^\mu$ as there is only syntactic fixed point alternation of depth 1. Thus each of these formulas also belongs to $\Sigma_2^\mu \cap \Pi_2^\mu$. Together we have that both γ'_0 and each $\sigma Z.\varphi(Z)$ belongs to Σ_2^μ and so

$$\gamma'_0(Y_1/\sigma Z_1.\varphi_1(Z_1))...(Y_s/\sigma Z_s.\varphi_s(Z_s)) \in \Sigma_2^\mu$$

By the same argument

$$\gamma'_0(Y_1/\sigma Z_1.\varphi_1(Z_1))...(Y_s/\sigma Z_s.\varphi_s(Z_s) \in \Pi_2^\mu$$

and hence $\gamma_0 \in \Sigma_2^\mu \cap \Pi_2^\mu$. As γ is a Boolean combination of formulas like γ_0 and formulas that have no syntactic fixed point alternation, we conclude that $\gamma \in \Sigma_2^\mu \cap \Pi_2^\mu$. Therefore the constructed interpolant lies in the alternation free fragment of the modal mu-calculus. □

[3] Notice that so far we talked about syntactic fixed point alternation depth, which is not the same as the defined alternation depth in chapter 2. If a formula has syntactic fixed point alternation 2, it belongs to $\Sigma_2^\mu \cup \Pi_2^\mu$ and can therefore also belong to $\Sigma_2^\mu \cap \Pi_2^\mu$. If a formula has alternation depth 2, then it belongs to $\Sigma_3^\mu \cap \Pi_3^\mu$ and can therefore not belong to $\Sigma_2^\mu \cap \Pi_2^\mu$, as the alternation depth hierarchy is strict.

Chapter 7

Discussion and further research

This thesis contributes to our proof theoretic understanding of the modal mu-calculus, in particular of the first level of the alternation depth hierarchy. The main contribution is the construction of a circular tableaux and a circular proof system for $\Sigma_1^\mu \cup \Pi_1^\mu$. With these two systems we do not only provide a proper axiomatization of $\Sigma_1^\mu \cup \Pi_1^\mu$, but we also establish that this fragment enjoys both the finite model property and Craig interpolation. The circular systems rely on the lack of proper fixed point alternation in formulas belonging to $\Sigma_1^\mu \cup \Pi_1^\mu$. For sequents containing such formulas it is relatively straightforward to identify regular structures in infinitary proof trees or tableaux. For example, in order to identify non-axiomatic leafs of a circular proof, it suffices to build a respective branch of a proof tree up to the first repetition containing a Π_1^μ-formula on the right or a Σ_1^μ-formula on the left. The existence of a circular proof system for the first level of the alternation hierarchy yields the question how to extend our work to obtain circular systems for the alternation free fragment? We suspect that this extension is relatively straightforward, as the alternation free fragment contains no formulas with proper fixed point alternation. Closely related to this is the question how to extend our work to obtain circular proof systems for modal logics with common knowledge? Both questions remain open for now, but we hope to answer them in the near future.

The fact that the first level of the alternation depth hierarchy enjoys the finite model property does not come as a surprise. Emerson and Streett [17] already established in 1989 that the whole modal mu-calculus enjoys the finite model property. Our result is in that sense a corollary of the general case. However, we do provide a novel proof of it using circular tableaux, instead of the automata theoretic approach of Emerson and Streett. Moreover, our proof is considerably easier than the proof for the whole mu-calculus and we hope that it provides new insight into the fact that formulas from the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$ which are satisfiable, do have a finite model. Furthermore, our proof provides a relatively easy way to obtain such a finite model. Given a satisfiable formula, we build a circular tableaux for it and directly read of the finite model for the formula. The constructed circular tableaux system in chapter

4 also provide us with a decision procedure for checking satisfiability. Let us quickly sketch this procedure. A proper treatment is left for future research. Recall the definition of a tableaux in normal form. This notion is easily adjusted to circular tableaux in normal form, where we require to identify the first possible repetition to be an non-axiomatic leaf. Given a formula $\varphi \in \Sigma_1^\mu \cup \Pi_1^\mu$, a circular pre-tableau in normal form for φ with respect to an arbitrary enumeration of the subformulas of φ is determined up to disjunctions. If φ does not contain a disjunction, then there is only a single circular pre-tableau for φ which is normal with respect to the enumeration. If φ contains n disjunction symbols, then there are at most 2^n different circular pre-tableaux in normal form. Moreover, recall that we require branches through circular pre-tableaux in normal form to end as soon as they reach the first suitable repetition for a non-axiomatic leaf. That is, every branch ends after at most $2^{|Sub(\varphi)|}$-many nodes. Thus, when building a circular pre-tableaux, we can stop the construction of each branch, if after $2^{|Sub(\varphi)|}$-many nodes no suitable axiomatic or non-axiomatic leaf has been reached. Finally, the result that a formula which has tableau, has a tableau in normal form with respect to an arbitrary enumeration can be established for circular tableaux as well. Therefore, in order to check whether φ is satisfiable, we can fix an arbitrary enumeration of the subformulas of φ and then algorithmically check every possible circular pre-tableaux for φ with respect to the enumeration. As the number of such pre-tableaux is exponential in the number of disjunctions occurring in φ, this algorithm always terminates. This gives us the result that the satisfiability problem for $\Sigma_1^\mu \cup \Pi_1^\mu$

Given a closed formula $\varphi \in \Sigma_1^\mu \cup \Pi_1^\mu$, is it satisfiable?

is decidable. Unfortunately, the complexity of the algorithm is clearly atleast EXPTIME, which is not a progress compared to the satisfiability problem for the whole modal mu-calculus, which is EXPTIME-complete. An open question is therefore whether this upper bound can be optimized and if yes, how to do so?

Similar to the finite model property, Craig interpolation has also already been established for the whole modal mu-calculus by D'Agostino and Hollenberg [7] in 2000. Our main contribution is a novel proof for the fragment $\Sigma_1^\mu \cup \Pi_1^\mu$ using a circular proof system. Particularly interesting is the negative result, stating that given an arbitrary circular proof for a sequent $\Gamma \Rightarrow \Delta$, the constructed Craig interpolant does in general belong to the alternation free fragment of the modal mu-calculus and is therefore not optimal. This is caused by circular proof trees that include branching into at least three branches such that two of them are only labelled by formulas on one side. Moreover, the labelled sides are opposite of each other. The circular proof in example 6.5.1 shows such a case. However, we also noticed that the sequent in example 6.5.1 has another circular proof for which the constructed interpolant belongs to the closure of $\Sigma_1^\mu \cup \Pi_1^\mu$. This illustrates that the proof tree at hand has a crucial impact on the complexity of the constructed interpolant. It remains an open question whether every valid sequent has a circular proof, such that no syntactic fixed point alternation is introduced into the interpolant. In case this question has a negative answer, it remains to show to what degree this phenomena would be caused by the specific proof system at hand? We leave these questions for further research.

Apart from circular proof systems we also provide infinitary sequent calculi, of which the calculus $\mathbf{DT}'$ and the calculus $\mathbf{2DT}$ are the most natural. The calculus $\mathbf{DT}'$ is essentially the sequent calculus T_μ^{pre} discussed by Studer in [18]. The only difference are the fixed point rules. The fixed point rules of T_μ^{pre} is a combination of the rule (σ) where $\sigma \in \{\mu, \nu\}$ and Z of $\mathbf{DT}'$ into the following rule:

$$\frac{\Gamma, \varphi(\sigma Z.\varphi(Z))}{\Gamma, \sigma Z.\varphi(Z)} \; (\sigma)_\mu^{pre}$$

It is straightforward to show that replacing our fixed point rules with this rule results in an equivalent sequent calculus. The rule $(\sigma)_\mu^{pre}$ enjoys the advantage that it generates no formulas that contain free variables. Using the rules (σ) and (Z) results in formulas that label the nodes of a proof tree and do indeed have free variables. Clearly, these free variables are interpreted to be bound, but technically they occur free. This causes problems for example in the soundness proof of $\mathbf{DT}$, where we have to replace every negated variable by a non-negated variable in the translation $D(\cdot)$. The advantage of our rules is that they result in a nicer presentation of proof trees. Given a long formula of the form $\sigma Z.\varphi(Z)$, the formula $\varphi(\sigma Z.\varphi(Z))$ can be become incredibly nasty. The infinitary sequent calculi discussed in this thesis are therefore not essentially new. However, we do provide novel soundness and completeness proofs by using the method of translation from the infinitary tableaux system $\mathbf{T}$. The soundness and completeness proof of $\mathbf{DT}$ moreover provides insight into the connections between tableaux systems and proof systems.

Bibliography

[1] B. Afshari and G. E. Leigh. Cut-free completeness for modal mu-calculus. In *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–12, 2017.

[2] Bahareh Afshari. Logic, games and automata (lecture notes). University of Amsterdam, 2020.

[3] Bahareh Afshari and Graham E. Leigh. Lyndon interpolation for modal mu-calculus. In *Logic, Language, and Computation, Post-proceedings of the 13th International Tbilisi Symposium on Logic, Language, and Computation*. Lecture Notes in Computer Science, Springer, Forthcoming.

[4] Julian Bradfield. The modal mu-calculus alternation hierarchy is strict. *Theoretical Computer Science*, 195(2):133 – 153, 1998. Concurrency Theory.

[5] Julian Bradfield and Colin Stirling. 12 modal mu-calculi. In Patrick Blackburn, Johan Van Benthem, and Frank Wolter, editors, *Handbook of Modal Logic*, volume 3 of *Studies in Logic and Practical Reasoning*, pages 721 – 756. Elsevier, 2007.

[6] Giovanna D'Agostino. Interpolation in non-classical logics. *Synthese*, 164(3):421–435, 2008.

[7] Giovanna D'Agostino and Marco Hollenberg. Logical questions concerning the mu-calculus. *The Journal of Symbolic Logic*, 65(1):310–332, 2000.

[8] Stéphane Demri, Valentin Goranko, and Martin Lange. *Temporal Logics in Computer Science: Finite-State Systems*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2016.

[9] Erich Grädel. Why are modal logics so robustly decidable? In Gheorghe Paun, Grzegorz Rozenberg, and Arto Salomaa, editors, *Current Trends in Theoretical Computer Science, Entering the 21th Century*, pages 393–408. World Scientific, 2001.

[10] Gerhard Jäger, Mathis Kretz, and Thomas Studer. Canonical completeness of infinitary mu. *The Journal of Logic and Algebraic Programming*, 76(2):270 – 292, 2008. Logic and Information: From Logic to Constructive Reasoning.

[11] Natthapong Jungteerapanich. A tableau system for the modal mu-calculus. In Martin Giese and Arild Waaler, editors, *Automated Reasoning with Analytic Tableaux and Related Methods*, pages 220–234, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[12] Dexter Kozen. Results on the propositional mu-calculus. *Theoretical Computer Science*, 27(3):333 – 354, 1983. Special Issue Ninth International Colloquium on Automata, Languages and Programming (ICALP) Aarhus, Summer 1982.

[13] Donald A. Martin. Borel determinacy. *Annals of Mathematics*, 102(2):363–371, 1975.

[14] David E. Muller, Ahmed Saoudi, and Paul E. Schupp. Alternating automata, the weak monadic theory of trees and its complexity. *Theoretical Computer Science*, 97(2):233 – 244, 1992.

[15] Damian Niwiński. On fixed-point clones. In Laurent Kott, editor, *Automata, Languages and Programming*, pages 464–473, Berlin, Heidelberg, 1986. Springer Berlin Heidelberg.

[16] Damian Niwiński and Igor Walukiewicz. Games for the mu-calculus. *Theoretical Computer Science*, 163(1):99–116, 1996.

[17] Robert S. Streett and E. Allen Emerson. An automata theoretic decision procedure for the propositional mu-calculus. *Information and Computation*, 81(3):249 – 264, 1989.

[18] Thomas Studer. On the proof theory of the modal mu-calculus. *Studia Logica: An International Journal for Symbolic Logic*, 89(3):343–363, 2008.

[19] Alfred Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific J. Math.*, 5(2):285–309, 1955.