Saarland University
Faculty of Natural Sciences and Technology I
Department of Computer Science

Bachelor's Program in Computer Science

Bachelor's Thesis

Terminating Tableaux for Modal Logic with Transitive Closure

submitted by

Sigurd Ernst Walter Schneider

on August, 21st, 2009

Supervisor

Prof. Dr. Gert Smolka

Advisors

Mark Kaminski, M.Sc.
Prof. Dr. Gert Smolka

Reviewers

Prof. Dr.-Ing. Holger Hermanns
Prof. Dr. Gert Smolka

Statement in Lieu of an Oath

I hereby confirm that I have written this thesis on my own and that I have not used any other media or materials than the ones referred to in this thesis.

Declaration of Consent

I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department.

Saarbrücken, ____________________ ____________________

Date Signature

Abstract

We present a terminating tableau system for the modal logic K*. K* extends the basic modal logic K with a reflexive transitive closure operator for relations and is a proper fragment of propositional dynamic logic.

We investigate two different approaches to achieve termination, namely chain-based blocking and pattern-based blocking. Pattern based-blocking has not been applied to a modal logic with a reflexive transitive closure operator.

We have a modular completeness proof that adapts to both termination approaches. Extending completeness arguments for a related description logic, we establish a strengthened soundness property of our calculus that we call straightness. Using this property we are able to prove both verification and refutation soundness.

Acknowledgment

My deepest gratitude is due to Mark Kaminski and Gert Smolka for offering me such an interesting topic of investigation. Their explanations were invaluable for my understanding of the subject matter, and from the continuous discussions I have profited far beyond the scope of this thesis.

I would also like to thank Holger Hermanns for reviewing this thesis, and Bernd Finkbeiner for his helpful advice and support in all academic undertakings.

I am deeply grateful to my family and Christin for their continued support in all circumstances, their understanding, and their advice on questions mathematics cannot answer.

Contents

1 Introduction 7
1.1 Contributions . . . 8
1.2 Outline . . . 9

2 Fundamentals 10
2.1 Syntax . . . 10
2.1.1 Substitution . . . 11
2.1.2 Names . . . 11
2.2 Semantics . . . 11
2.2.1 Locality . . . 11
2.2.2 Defining Equations . . . 11
2.2.3 Transitive Closure Operator . . . 12
2.2.4 Satisfiability . . . 12
2.2.5 Modally Valid Formulas . . . 13

3 Models 14
3.1 Relational Structure . . . 14
3.2 Labels in Models . . . 14
3.3 Requests . . . 14
3.4 Canonical Models . . . 15

4 Tableau System 19
4.1 Branches . . . 19
4.1.1 Labels . . . 20
4.1.2 Patterns . . . 20
4.1.3 Relations . . . 21
4.1.4 Paths . . . 21
4.2 Model Existence . . . 22
4.2.1 Evidence Conditions . . . 22
4.2.2 Model Existence Theorem . . . 23
4.3 Rules . . . 25
4.4 The Basic System $\mathcal{T}$ . . . 25
4.5 Admissibility . . . 27
4.6 Soundness . . . 28
4.7 Straightness . . . 28
4.7.1 Witness Distances . . . 29
4.7.2 Selectors . . . 29

4.7.3 Straight Branches 30
4.7.4 Straightness of $\mathcal{T}$ 31
4.7.5 Path Variables and Straightness 32
4.7.6 Straight Models 32

5 Chain-based Control **34**
5.1 Blocking 34
5.1.1 The Restricted System $\mathcal{T}_{chn}$ 34
5.2 Termination 35
5.2.1 Expressions 35
5.2.2 Variables 36
5.2.3 Termination Proof 36
5.3 Maximal Tableaux 37
5.3.1 Closed Branches 37
5.3.2 Maximal, Open Branches 37
5.3.3 Maximal, Open and Evident Branches 38
5.4 Completeness of $\mathcal{T}_{chn}$ 39

6 Pattern-based Control **41**
6.1 Blocking 41
6.1.1 $\Diamond$-Patterns 41
6.1.2 Realization of $\Diamond$-Patterns 41
6.1.3 The Restricted System $\mathcal{T}_{pat}$ 42
6.1.4 Remark: Pattern Inclusion 42
6.2 Termination 43
6.2.1 Patterns 43
6.2.2 Formulas 43
6.2.3 Termination Proof 44
6.3 Maximal Tableaux 45
6.3.1 Closed Branches 45
6.3.2 Maximal, Open Branches 45
6.3.3 Maximal, Open, and Evident Branches 47
6.3.4 Detecting Loops 48
6.4 Completeness of $\mathcal{T}_{pat}$ 49

7 Conclusion **51**

References **55**

1 Introduction

In this thesis we present two terminating tableau systems for a modal logic we call K^*. K^* enriches the modal logic K with a reflexive transitive closure operator for relations. The expressiveness of K^* lies between that of the basic modal logic K and propositional dynamic logic, of which K^* is a proper fragment.

Modal logic was treated deductively since 1918 [25], and received its relational semantics around 1960. The most significant contributions are from Kripke [22, 23, 24], and we refer to [10] for a detailed overview. Since then, relational semantics matured from an analytical tool to the standard way of thinking about modal logic.

Relational semantics make the concept of *locality* of truth explicit: The notion of truth depends on the state of evaluation. This allows the natural modeling of computation as state transition. On the basis of this concept, dynamic logic provides means to specify programs and reason about their properties.

Dynamic logic is due to Fisher and Ladner [8]. Their work extends Pratt's [26], who was in search of a semantics for Hoare logic [14]. Hoare logic abstracts computation as transition between program states, thus committing to programs that are binary relations. Dynamic Logic subsumes Hoare logic and additionally allows programs to be represented *within* the language. Having both programs and properties represented in the language is referred to as the exogenous approach [13].

In dynamic logic, programs *are* relations, and the language provides several operators to compose new programs out of old ones. For two reasons, the reflexive transitive closure operator is the most challenging of the program composition operators. First, defining the reflexive transitive closure over relations requires higher order quantification. And second, adding the reflexive transitive closure operator to obtain K^* results in the loss of compactness of the logic. Nevertheless, K^* is decidable and possesses the small model property. Both properties are inherited from PDL, for which they are shown by Pratt in 1976 [26] and Fischer and Ladner [8] in 1977, respectively.

Proving completeness for a terminating tableau system for K^* inherits a difficulty from PDL: There are derivations for unsatisfiable formulas such that no finite prefix of the derivation contains an immediate contradiction. This infinitary behavior results from the way the transitive closure operator is treated and the lack of compactness. For terminating tableau systems, this is a concern of soundness, since a derivation without a contradiction cannot be discarded without justification.

In the literature, there exist different tableau systems for PDL. An early approach by Pratt [26] resorts to a weaker semantics and annotates the tableau to keep track of fulfilled eventualities. De Giacomo and Massacci [9] pick up this idea and show completeness by carrying an interpretation along the derivation that satisfies the generated formulas. Their decision procedure features path variables, which ease the tracking of unfulfilled eventualities greatly. Baader [3] treats completeness for the description logic $\mathrm{ALC}_{\mathrm{trans}}$, which subsumes K^*. In his approach, an interpretation is used to guide a derivation, featuring an elaborate technique to ensure invariants on the derivation. These invariants are then used to prove verification completeness. Abate, Goré and Widmann [1] are extending Pratt's approach to an on-the-fly procedure. They incrementally build an and-or-graph and propagate information about unfulfilled eventualities upwards, such that properties of the root of the tree indicate satisfiability. Goré and Widmann [11] recently published an optimal tableau decision procedure.

Our approach incorporates ideas of Baader, De Giacomo, and Massacci. We show a property similar to soundness for our calculus. This property is called straightness and guarantees preservation of certain invariants during the derivation process. These invariants enable us to prove a model existence theorem for maximal derivations that satisfy them, in particular we show that such derivations satisfy all eventualities. We justify discarding derivations by establishing that the straightness properties are violated and give conditions to identify such derivations.

To obtain a terminating tableau system, rule applicability must be restricted. We call a technique that provides conditions to control rule applicability a blocking technique. We equip our system with two different blocking techniques and show completeness as described above for both of them.

First, we equip the tableau with chain-based blocking. Chain-based blocking is the traditional blocking technique for modal logics invented by Kripke [23]. It is standard and has been applied to a variety of modal and description logics [5, 15, 16, 18]. The basic idea is that expansion can be blocked if a state that has the same properties as one of its predecessors is encountered. Instead of creating a new state, the edges can be redirected to the corresponding predecessor.

Second, we equip the tableau system with pattern-based blocking. Pattern-based blocking is a recent idea due to Kaminski and Smolka [19, 20], which has been applied successfully to various logics [21, 17]. Spartacus [12], a maturing implementation incorporating the technique, shows promising performance results. Patterns are subsets of the formulas present at a certain nominal. The main idea is that similar patterns require similar sub-models. Thus similar patterns are identified, and expansion is blocked for all instances, except for one, for which a sub-model is generated. Model existence is achieved by sharing this sub-model among all states that correspond to a nominal with a similar pattern. The precise notion of similarity depends on what the logic admits, but usually inclusion or equality applies.

We conclude the thesis with a conjecture about canonical models, and give an outlook on possible improvements and future work.

1.1 Contributions

The main contribution of this thesis is the adaptation of pattern-based blocking to a logic with the reflexive transitive closure operator. Our completeness proof is more modular than existing proofs for PDL [3, 9] in that it adapts to two both chain-based and pattern-based blocking.

1.2 Outline

In Chapter 2, we define syntax and semantics for K^*. In Chapter 3, we prove decidability and the small model property for K^* using a canonical model construction. In Chapter 4, we give a tableau calculus, prove a model existence theorem, show that the system is sound, and establish our extended soundness property called straightness. In Chapter 5, we equip the tableaux system with chain-based blocking, prove termination, devise conditions to identify discardable derivations, and prove completeness of the system. In Chapter 6, we do the same for pattern-based blocking.

2 Fundamentals

In this chapter, we define the syntax and semantics of K* as a fragment of simple type theory (STT) [6, 2]. In Section 2.1, we define the syntax of K*. In Section 2.2, we define the semantics using STT and give a definition of satisfiability.

2.1 Syntax

In Fig. 2.1 the syntactic categories of modal expressions and relations are given. For the sake of simplicity we only consider negation normal terms. The names $\dot\neg$, $\dot\wedge$, $\dot\vee$, $\Diamond$, $\Box$, and * are called **modal constants**. Note that we use infix notation for the operators $\dot\wedge$ and $\dot\vee$, as well as postfix notation for the operator *.

Figure 2.1: Syntactic Categories of K^*

$t \ ::= \ p \mid \dot\neg t \mid t \,\dot\wedge\, t \mid t \,\dot\vee\, t \mid \Diamond\rho t \mid \Box\rho t$	**modal expressions**
$\rho ::= \ r \mid r^*$	**relations**

Often it is useful to have the subterm closure at hand. Note that we here have a extended subterm closure, particularly K*'s version of the Fischer-Ladner closure [8].

Definition 1 Let t be a modal expression. Then $\mathcal{M}t$ is inductively defined according to the following equations.

$$\begin{aligned}
\mathcal{M}p &= \{p\} \\
\mathcal{M}(\dot\neg t) &= \{\dot\neg t\} \cup \mathcal{M}t \\
\mathcal{M}(t_1 \,\dot\wedge\, t_2) &= \{t_1 \,\dot\wedge\, t_2\} \cup \mathcal{M}t_1 \cup \mathcal{M}t_2 \\
\mathcal{M}(t_1 \,\dot\vee\, t_2) &= \{t_1 \,\dot\vee\, t_2\} \cup \mathcal{M}t_1 \cup \mathcal{M}t_2 \\
\mathcal{M}(\Diamond rt) &= \{\Diamond rt\} \cup \mathcal{M}t \\
\mathcal{M}(\Diamond r^*t) &= \{\Diamond r(\Diamond r^*t)\} \cup \mathcal{M}t \\
\mathcal{M}(\Box\rho t) &= \{\Box\rho t\} \cup \mathcal{M}t
\end{aligned}$$

Let L be a set of modal expressions. Then we define $\mathcal{M}L := \bigcup\{\mathcal{M}t \mid t \in L\}$.

Proposition 1 Let t be a modal expression. Then $\mathcal{M}t$ is a finite set.

2.1.1 Substitution

Given a term t, we use the notation t^x_y to indicate the term obtained from t by subsituting y for x. Given a set A, the notation A^x_y indicates the set obtained by substituting y for x in every element of A. A detailed treatment of substitution can be found in [27].

2.1.2 Names

We distinguish between **variables** and **constants**. The set of **constants** consists of the logical constants $\neg, \wedge, \vee, \rightarrow$, the quantifiers $\forall, \exists$ and the modal constants $\dot\neg, \dot\wedge, \dot\vee, \Diamond, \Box, ^*$. We use $\mathcal{N}$ to denote the set of all names that are not constants, and refer to it as **variables**. To denote the set of all variables of a certain type σ, we write $\mathcal{N}_\sigma$. Note that $\mathcal{N}_\iota$ denotes the set of all nominals. We divide $\mathcal{N}_{\iota o}$ in two disjoint subsets, namely **atomic properties** and **path variables**. We denote the set of all path variables by $\mathcal{V}$. We reserve the following letters for variables of the given type:

$$
\begin{aligned}
x, y, z &: \iota \\
p, q &: \iota o \qquad\qquad p, q \notin \mathcal{V} \\
\alpha, \beta &: \iota o \qquad\qquad \alpha, \beta \in \mathcal{V} \\
r &: \iota\iota o
\end{aligned}
$$

2.2 Semantics

We now approach semantics by embedding modal logic into Simple Type Theory (STT) to have its rich syntactic and semantic framework at hand. For an introduction to STT refer to [2]. To know more, read [7].

We start from two base types o and ι and fix the interpretation of o to be the set $\{0, 1\}$. The interpretation of ι is a non-empty set of **states**. Given two types σ and τ, the interpretation of the type $\sigma\tau$ is the set of all total functions from the interpretation of σ to the interpretation of τ. Types are right-associative.

We use names of type ι as states and call them **nominals**, names of type ιo as labels and call them **atomic properties**, and names of type $\iota\iota o$ as **atomic relations**.

2.2.1 Locality

Modal expressions are always of type ιo, whereas **modal formulas** are of type o. Note that traditional modal formulas [4] correspond to the modal expressions in our setup. Intuitively, going from a modal expression t to a modal formula tx augments the property denoted by t with the state of evaluation.

2.2.2 Defining Equations

The embedding into STT is straightforward and has been successfully done before. See for example [20, 21]. A listing of the defining equations for the modal constants is given in Fig. 2.2.

Figure 2.2: Defining Equations for the Traditional Modal Constants

$\dot{\neg} = \lambda px.\neg px$	$\dot{\neg} : (\iota o)\iota o$
$\dot{\wedge} = \lambda pqx.\ px \wedge qx$	$\dot{\wedge} : (\iota o)(\iota o)\iota o$
$\dot{\vee} = \lambda pqx.\ px \vee qx$	$\dot{\vee} : (\iota o)(\iota o)\iota o$
$\Diamond = \lambda rpx.\ \exists y.\ rxy \wedge py$	$\Diamond : (\iota\iota o)(\iota o)\iota o$
$\Box = \lambda rpx.\ \forall y.\ rxy \rightarrow py$	$\Box : (\iota\iota o)(\iota o)\iota o$

2.2.3 Transitive Closure Operator

We now define the reflexive transitive closure operator in STT. We define the reflexive transitive closure of a relation r as the intersection of all reflexive transitive relations that contain r.

Definition 2 Let r be a relation. The reflexive, transitive closure r^* of r is defined as follows.

$$r^* := \bigcap \{r' \mid r' \text{ reflexive} \wedge r' \text{ transitive} \wedge r \subseteq r'\}$$

This equation is easy to express in STT. Note that we obtain the infinite intersection by using an universal quantifier over relations of type $\iota\iota o$.

Figure 2.3: Defining Equations for the Transitive Closure Operator

$\mathrm{T} = \lambda r.\ \forall xyz.rxy \wedge ryz \rightarrow rxz$	$\mathrm{T} : (\iota\iota o)o$
$\mathrm{R} = \lambda r.\ \forall x.rxx$	$\mathrm{R} : (\iota\iota o)o$
$\subseteq\ = \lambda rr'.\ \forall xy.rxy \rightarrow r'xy$	$\subseteq : (\iota\iota o)(\iota\iota o)o$
$^* = \lambda rxy.\ \forall r'.\mathrm{R}r' \wedge \mathrm{T}r' \wedge r \subseteq r' \rightarrow r'xy$	$^* : (\iota\iota o)\iota\iota o$

2.2.4 Satisfiability

A **modal interpretation** is an interpretation of simple type theory that interprets o as the set $\{0, 1\}$, the logical constants and quantifiers as usual, and the modal constants according to their definition.

Definition 3 A modal interpretation $\mathcal{I}$ **satisfies a set of modal formulas** A, if $\mathcal{I} \models A$, i.e. $\mathcal{I}s = 1$ for all $s \in A$. A modal interpretation $\mathcal{I}$ **satisfies a set of modal expressions** L, if there is a state $a \in \mathcal{I}\iota$ such that $\mathcal{I}, a \models L$, i.e. $\mathcal{I}ta = 1$ for all $t \in L$.

We often write $\mathcal{I} \models s$, where s is a modal formula. In this case we mean the singleton set containing s. The same applies to modal expressions.

We import a proposition from [20].

Proposition 2 Let $\mathcal{I}$ be an interpretation and X be a set of nominals such that $\mathcal{I}\iota = \{\mathcal{I}x \mid x \in X\}$. Then the following holds for every term t of type ιo:

$$\begin{aligned} \mathcal{I} \models \exists t &\iff \exists x \in X\colon \mathcal{I} \models tx \\ \mathcal{I} \models \forall t &\iff \forall x \in X\colon \mathcal{I} \models tx \end{aligned}$$

Since the above proposition is not applicable to all interpretations, we have another proposition to lift quantifiers to the meta level.

Proposition 3 Let $\mathcal{I}$ be an interpretation. Then the following holds for every term t of type ιo:

$$\begin{aligned} \mathcal{I} \models \exists t &\iff \exists a \in \mathcal{I}\iota\colon \mathcal{I}, a \models t \\ \mathcal{I} \models \forall t &\iff \forall a \in \mathcal{I}\iota\colon \mathcal{I}, a \models t \end{aligned}$$

We also have a proposition concerned with the semantics of $\Box$ and $\Diamond$.

Proposition 4 Let $\mathcal{I}$ be a modal interpretation and $a \in \mathcal{I}\iota$. Then the following holds.

$$\begin{aligned} \mathcal{I}, a \models \Diamond\rho t &\iff \exists b \in \mathcal{I}\iota\colon \mathcal{I}\rho ab = 1 \wedge \mathcal{I}, b \models t \\ \mathcal{I}, a \models \Box\rho t &\iff \forall b \in \mathcal{I}\iota\colon \mathcal{I}\rho ab = 1 \implies \mathcal{I}, b \models t \\ \mathcal{I}, a \models \Box r^* t &\implies \forall b \in \mathcal{I}\iota\colon \mathcal{I}rab = 1 \implies \mathcal{I}, b \models t, \Box r^* t \end{aligned}$$

2.2.5 Modally Valid Formulas

A formula is **modally valid**, if it is satisfied by all modal interpretations. We give some important modally valid formulas. They can be verified by any tableaux system for STT using the defining equations for the modal constants from Fig. 2.2. Suitable tableau systems can be found in [27, 2], for example.

$$\begin{aligned} \Diamond\rho p &= \dot{\neg}\Box\rho(\dot{\neg}p) \\ \Box r^* px &\equiv px \wedge \Box r(\Box r^* p)x \\ \Diamond r^* px &\equiv px \vee \Diamond r(\Diamond r^* p)x \end{aligned}$$

3 Models

In this chapter, we establish the existence of a class of models for K*, which we call canonical models. In Section 3.1, we define notation to reason about the relational structure encoded in modal interpretations. In Section 3.2, we define a notion of labels of a state, which is a subset of the properties the state satisfies. In Section 3.3, we compute from the labels of a state a set of formulas every successor of the state must satisfy. In Section 3.4, we construct for every satisfiable K*-expression a finite, satisfying model. We get decidability and the small model property as corollaries.

3.1 Relational Structure

Since modal interpretations encode a relational structure, it is convenient to have an explicit way to talk about related nominals.

Definition 4 Let $\mathcal{I}$ be a modal interpretation, $a, b \in \mathcal{I}\iota$ and r be a relation.

$$a \xrightarrow{r}_{\mathcal{I}} b :\iff \mathcal{I}rab = 1$$

Proposition 5 Let $\mathcal{I}$ be a modal interpretation, and $\longrightarrow \in \mathcal{I}\iota \times \mathcal{I}\iota$ be a relation such that $\longrightarrow = \mathcal{I}r$. Then for all $a, b \in \mathcal{I}\iota$ the following holds:

$$\mathcal{I}r^*ab = 1 \iff a \longrightarrow^* b$$

3.2 Labels in Models

Every state in a model satisfies an infinite number of modal expressions. To work with a reasonable set of modal expressions, we define the **labels of a state** with respect to a set of modal expressions as follows.

Definition 5 Let $\mathcal{I}$ be a modal interpretation, $a \in \mathcal{I}\iota$, and L be a set of modal expressions.

$$\mathcal{L}^L_{\mathcal{I}}a := \{t \in L \mid \mathcal{I}, a \models t\}$$

3.3 Requests

Let L be a set of modal expressions. The **r-request** of L is the set of modal expressions every r-successor of a state satisfying L must inevitably satisfy. An analogous concept is used by Goré et al [1, 11].

Definition 6 We denote the request of a set of modal expressions with $\mathcal{R}^r L$.

$$\mathcal{R}^r L := \{t \mid \Box rt \in L\} \cup \{t, \Box r^* t \mid \Box r^* t \in L\}$$

We call $\mathcal{R}^r(\mathcal{L}^L_{\mathcal{I}} a)$ the **request of the state *a*** in $\mathcal{I}$ with respect to L.

Proposition 6 Let $\mathcal{I}$ be a modal interpretation, L be a set of modal expressions, $a, b \in \mathcal{I}\iota$ such that $a \xrightarrow{r}_{\mathcal{I}} b$, and $\mathcal{I}, a \models L$. Then $\mathcal{I}, b \models \mathcal{R}^r L$.

Proof. We prove $\Box rt \in L \implies \mathcal{I}, b \models t$ and $\Box r^* t \in L \implies \mathcal{I}, b \models t, \Box r^* t$.

Let $\Box rt \in L$.

$$\begin{aligned} & \mathcal{I}, a \models \Box rt \\ \iff & \forall b \in \mathcal{I}\iota \colon \mathcal{I}rab = 1 \implies \mathcal{I}, b \models t && \text{Prop. 4} \\ \implies & \mathcal{I}, b \models t && \mathcal{I}rab = 1 \end{aligned}$$

Let $\Box r^* t \in L$.

$$\begin{aligned} & \mathcal{I}, a \models \Box r^* t \\ \iff & \forall b \in \mathcal{I}\iota \colon \mathcal{I}rab = 1 \implies \mathcal{I}, b \models t \land \mathcal{I}, b \models \Box r^* t && \text{Prop. 4} \\ \implies & \mathcal{I}, b \models t \land \mathcal{I}, b \models \Box r^* t && \mathcal{I}rab = 1 \end{aligned}$$

□

Lemma 1 Let L be a satisfiable set of modal expressions. If $\Diamond rt \in L$ then $(\mathcal{R}^r L) \cup \{t\}$ is satisfiable.

Proof. Since L is satisfiable, there is an interpretation $\mathcal{I}$ and a state $a \in \mathcal{I}\iota$ such that $\mathcal{I}, a \models A$.

$$\begin{aligned} & \mathcal{I}, a \models L \\ \iff & \mathcal{I}, a \models \Diamond rt && \Diamond rt \in L, \text{Def. 3} \\ \iff & \exists b \in \mathcal{I}\iota \colon a \xrightarrow{r}_{\mathcal{I}} b \land \mathcal{I}, b \models t && \text{Prop. 4} \\ \implies & \exists b \in \mathcal{I}\iota \colon \mathcal{I}, b \models \mathcal{R}^r L \land \mathcal{I}, b \models t && \text{Prop. 6 with } \mathcal{I}, a \models L \\ \iff & \exists b \in \mathcal{I}\iota \colon \mathcal{I}, b \models \mathcal{R}^r L \cup \{t\} && \text{Def. 3} \\ \implies & \mathcal{R}^r A \cup \{t\} \text{ satisfiable} \end{aligned}$$

□

3.4 Canonical Models

We now define canonical models following the construction for PDL as it appears in [4]. However, our argument does not use a notion of consistency, but is based on the semantic notion of satisfiability. We thus need no proof system, which makes the argument in our view more transparent.

Definition 7 (Maximal Satisfiable Subset) Let S be a set of modal formulas. A set A is a **maximal satisfiable subset of S** if $A \subseteq S$, A is satisfiable, and all sets B such that $A \subsetneq B \subseteq S$ are not satisfiable.

Proposition 7 Let $A \subseteq S$ such that A is satisfiable. Then there is a maximal satisfiable subset B of S such that $A \subseteq B$.

Definition 8 (Saturation Closure) Let t be a modal expression. Then saturation closure $\mathcal{C}$ for t is defined as follows.

$$\mathcal{C}t := \mathcal{M}t \cup \{\dot{\neg} t \mid t \in \mathcal{M}t\}$$

Note that we assume $\mathcal{C}t$ to contain only negation normal terms.

Definition 9 (Canonical Model) Let t be a modal expression, and $\mathcal{I}$ be a modal interpretation. $\mathcal{I}$ is a **canonical model of t** if it satisfies the following equations.

$$\begin{aligned} \mathcal{I}\iota &:= \{A \mid A \text{ maximal satisfiable subset of } \mathcal{C}t\} \\ \mathcal{I}pA &:= p \in A \\ \mathcal{I}rAB &:= (\mathcal{R}^r A) \subseteq B \end{aligned}$$

We now prove that each state satisfies the formulas it contains.

Theorem 1 (Satisfiability) Let t be a modal expression and $\mathcal{I}$ be a canonical model of t. Then $A \in \mathcal{I}\iota \implies \mathcal{I}, A \models A$.

Proof. Let $s \in \mathcal{C}t$ and $\mathcal{N}$ be a set of names such that $\mathcal{I}\iota = \{\mathcal{I}x \mid x \in \mathcal{N}\}$. We proof by induction on the structure of s for all $A \in \mathcal{I}\iota$ that $s \in A \implies \mathcal{I}, A \models s$. Let $A \in \mathcal{I}\iota$ and $s \in A$.

Let $p \in A$. $p \in A \iff \mathcal{I}pA = 1$ by Def. 9. Thus $\mathcal{I}, A \models p$.

Let $\dot{\neg} p \in A$. $\dot{\neg} p \in A \iff p \notin A$ by Def. 7 and $p \notin A \iff \mathcal{I}pA = 0$ by Def. 9.

Let $t_1 \dot{\wedge} t_2 \in A$. Let $x \in \mathcal{N}_\iota$ such that $\mathcal{I}x = A$.

$$\begin{aligned} t_1 \dot{\wedge} t_2 \in A &\iff t_1, t_2 \in A && \text{Def. 7} \\ &\implies \mathcal{I}, A \models t_1 \wedge \mathcal{I}, A \models t_2 && \text{Ind. Hyp.} \\ &\iff \mathcal{I} \models t_1 x \wedge \mathcal{I} \models t_2 x && \mathcal{I}x = A \\ &\iff \mathcal{I} \models t_1 x \wedge t_2 x && \\ &\iff \mathcal{I} \models (t_1 \dot{\wedge} t_2) x && \text{Fig. 2.2} \\ &\iff \mathcal{I}, A \models t_1 \dot{\wedge} t_2 && \mathcal{I}x = A \end{aligned}$$

Let $t_1 \dot\vee t_2 \in A$. Let $x \in \mathcal{N}_\iota$ such that $\mathcal{I}x = A$.

$$\begin{aligned} t_1 \dot\vee t_2 \in A &\iff t_1 \in A \vee t_2 \in A && \text{Def. 7} \\ &\implies \mathcal{I}, A \models t_1 \vee \mathcal{I}, A \models t_2 && \text{Ind. Hyp.} \\ &\iff \mathcal{I} \models t_1 x \vee \mathcal{I} \models t_2 x && \mathcal{I}x = A \\ &\iff \mathcal{I} \models t_1 x \vee t_2 x && \\ &\iff \mathcal{I} \models (t_1 \dot\vee t_2)x && \text{Fig. 2.2} \\ &\iff \mathcal{I}, A \models t_1 \dot\vee t_2 && \mathcal{I}x = A \end{aligned}$$

Let $\Diamond rt \in A$. Since A is satisfiable by Def. 9, we apply Lem. 1, and get that $(\mathcal{R}^r A) \cup \{t\}$ is satisfiable. By Prop. 7, there is maximal satisfiable subset B of $\mathcal{C}t$ containing $(\mathcal{R}^r A) \cup \{t\}$. B is a state in $\mathcal{I}$ by Def. 9. Again by Def. 9, we get $\mathcal{I}rAB = 1$ and the claim follows by induction for $t \in B$.

Let $s = \Diamond r^* t$. Since A is maximal satisfiable there is an interpretation $\mathcal{I}'$ with $a \in \mathcal{I}'\iota$ such that $\mathcal{I}', a \models A$. By Prop. 4 there is $b \in \mathcal{I}'\iota$ such that $a \xrightarrow{r}{}^{*}_{\mathcal{I}'} b$ and $\mathcal{I}'tb = 1$. There is $n \in \mathbb{N}$ such that $a \xrightarrow{r}{}^{n}_{\mathcal{I}'} b$. We show by induction on n that a path of the same length to a state satisfying t exists in $\mathcal{I}$.

- **Let** $n = 0$. Then $\mathcal{I}', a \models A \cup \{t\}$. Thus $t \in A$ since A is a maximal satisfiable subset of $\mathcal{C}t$ and $t \in \mathcal{C}t$. The claim follows by outer induction for $t \in A$.
- **Let** $n > 0$. We have $a \xrightarrow{r}_{\mathcal{I}'} c$, $c \xrightarrow{r}{}^{n-1}_{\mathcal{I}'} b$ for some $c \in \mathcal{I}'\iota$. We construct a maximal satisfiable subset of $\mathcal{C}t$, namely $B := \{t \mid t \in \mathcal{C}t \wedge \mathcal{I}tc = 1\}$. Observe that $B \in \mathcal{I}\iota$ by Def. 9. We also have that $\mathcal{I}', c \models \mathcal{R}^r A$ by Prop. 6, thus $\mathcal{R}^r A \subseteq B$, hence $\mathcal{I} \models rAB$. By construction $\mathcal{I}', c \models B$, thus we apply the inner induction hypothesis, which yields the claim.

Let $\Box rt \in A$. Let $x \in \mathcal{N}$ such that $\mathcal{I}x = A$.

$$\begin{aligned} &\forall y \in \mathcal{N} \colon \mathcal{I} \models rxy \leftrightarrow (\mathcal{R}^r A) \subseteq \mathcal{I}y && \text{Def. 9} \\ \iff &\forall y \in \mathcal{N} \colon \mathcal{I} \models rxy \to t \in \mathcal{I}y && t \in \mathcal{R}^r A \text{ by Def. 6} \\ \implies &\forall y \in \mathcal{N} \colon \mathcal{I} \models rxy \to \mathcal{I} \models ty && \text{Ind. Hyp.} \\ \iff &\forall y \in \mathcal{N} \colon \mathcal{I} \models rxy \to ty && \\ \iff &\forall y \in \mathcal{N} \colon \mathcal{I} \models (\lambda y.rxy \to ty)y && \beta\text{-law} \\ \iff &\mathcal{I} \models (\forall y.rxy \to ty) && \text{Prop. 2} \\ \iff &\mathcal{I} \models (\lambda rpx.\forall y.rxy \to py)rtx && \beta\text{-law} \\ \iff &\mathcal{I} \models \Box rtx && \text{Def. from Fig. 2.2} \\ \iff &\mathcal{I}, A \models \Box rt && \mathcal{I}x = A \end{aligned}$$

Let $\Box r^* t \in A$. Let $B \in \mathcal{I}\iota$ such that $A \xrightarrow{r}{}^{*}_{\mathcal{I}} B$. There is $n \in \mathbb{N}$ such that $A \xrightarrow{r}{}^{n}_{\mathcal{I}} B$. We show $\mathcal{I}, B \models t$ by induction on n.

Let $n = 0$. Since B is satisfiable by Def. 9 and $\Box r^* t \equiv t \dot\wedge \Box r(\Box r^* t)$ is a tautology, $t \in B$. The claim follows by outer induction.

Let $n > 0$. Then there is $C \in \mathcal{I}\iota$ such that $A \xrightarrow[\mathcal{I}]{r}{}^{1} C$ and $C \xrightarrow[\mathcal{I}]{r}{}^{n-1} B$. Thus $\mathcal{R}^r A \subseteq C$ by Def. 9, in particular $t, \Box r^* t \in C$. The claim follows by inner induction for $\Box r^* t \in C$. □

Having established this lemma, we can construct a canonical model for every modal expression t. If the expression is satisfiable, the canonical interpretation satisfies it.

Corollary 1 Let t be a modal expression and $\mathcal{I}$ be a canonical model of t. Then t is satisfiable if and only if $\mathcal{I}$ satisfies t.

The number of states in canonical models is bounded in the size of the saturation closure. Furthermore, the saturation closure is polynomially bounded in the size of the modal expression considered.

Corollary 2 (Small Model Property) Let t be a satisfiable modal expression. Then there is a modal interpretation $\mathcal{I}$ with $\mathcal{I}\iota \leq 2^{|\mathcal{C}t|}$ that satisfies t.

Having a bound on the size of the model of a formula, we can decide satisfiability of K*-formulas by enumerating all the models with a size below the bound and checking if any of them satisfies the formula.

Corollary 3 (Decidability) Satisfiability of modal expressions is decidable.

We have another interesting observation. This is essentially the backwards direction of Theorem 1.

Proposition 8 Let t be a modal expression, $\mathcal{I}$ be a canonical model for t, $a \in \mathcal{I}\iota$, $s \in \mathcal{C}t$, and $\mathcal{I}, a \models s$. Then $s \in a$.

Proof. Observe that there is an $s' \in \mathcal{C}t$ such that $s' = \dot\neg s$ by Def. 7. Assume $s \notin a$. Since a is a maximal satisfiable subset of $\mathcal{C}t$ by Def. 9, we have $s' \in a$. But then $\mathcal{I} \models \{\dot\neg s, s\}$ by Theorem 1, a contradiction. □

4 Tableau System

In this chapter, we will develop a sound and straight tableau system for K*. We precede as follows. In Section 4.1, we define the notion of a tableau branch and some important relations. In Section 4.2, we give a model existence theorem that establishes that branches satisfying certain conditions have a model. In Section 4.3, we give the tableau rules. In Section 4.4, we construct a basic (non-terminating) tableau system by applying the usual progress conditions. In Section 4.5, we formalize invariants on branches as admissibility conditions. In Section 4.6, we establish soundness of the system. In Section 4.7, we prove straightness properties for the system. Straightness is a strengthened soundness property, which asserts that the branch encodes a models of a certain class and thus has certain beneficial properties. The notion of encoding is given by the model existence theorem.

4.1 Branches

To talk about sets of formulas in the context of tableaux, we define a notion of **tableau branches**, or **branches** for short, as follows. Tableaux branches contain formulas obtained from the modal expressions t defined in Section 2.1, edges rxy, and formulas αx, $\Diamond r\alpha x$, and $\alpha = \Diamond r^* t$, the latter of which we call **equations**. In equations, the left hand side is always a path variable and the right hand side is always a $\Diamond^*$-formula. We denote the set of path variables occurring on a branch A by $\mathcal{V}A$.

To capture the special structure of the formulas that occur on tableau branches, we extend the grammar given in 2.1 by two new categories: the formulas s, and the extended terms u. We repeat the categories t and ρ in Def. 10 for convenience.

Definition 10 (Extended Grammar)

$$\begin{array}{rcll} s & ::= & ux \mid \Diamond r\alpha x \mid \alpha = \Diamond r^* t \mid rxx & \textbf{formula} \\ u & ::= & \alpha \mid t & \textbf{extended modal expression} \\ t & ::= & p \mid \dot{\neg} t \mid t \mathbin{\dot{\wedge}} t \mid t \mathbin{\dot{\vee}} t \mid \Diamond \rho t \mid \Box \rho t & \textbf{proper modal expression} \\ \rho & ::= & r \mid r^* & \end{array}$$

We use the letters s, u, t, ρ to range over the corresponding categories exclusively.

Adding new categories requires the extension of the function $\mathcal{M}$ to formulas.

Definition 11 Let s be an extended modal formula. We define $\mathcal{M}s$ inductively according to the following equations:

$$\begin{aligned}
\mathcal{M}(\alpha = \Diamond r^* t) &= \mathcal{M}(\Diamond r^* t) \\
\mathcal{M}(\Diamond r \alpha x) &= \{\Diamond r \alpha x, \alpha, x\} \\
\mathcal{M}(rxy) &= \{r, x, y\} \\
\mathcal{M}(tx) &= \{x\} \cup \mathcal{M}t \\
\mathcal{M}(\alpha x) &= \{\alpha, x\}
\end{aligned}$$

4.1.1 Labels

Definition 12 (Labels of a nominal) Let A be a branch and $x \in \mathcal{N}_\iota A$. We define the set of all **labels of a nominal x in A** as $\mathcal{L}_A x := \{t \mid tx \in A\} \cup \{\Diamond r^* t \mid \exists \alpha \in \mathcal{V}A\colon \alpha x, \alpha = \Diamond r^* t \in A\} \cup \{\Diamond r(\Diamond r^* t) \mid \alpha = \Diamond r^* t, \Diamond r \alpha x \in A\}$. A modal expression t is a **label** of x in A if $t \in \mathcal{L}_A x$.

Note that labels never contain path variables. Instead, they contain the corresponding modal expressions.

Definition 13 (Modal Equivalence) Let A be a branch. Two nominals $x, y \in \mathcal{N}_\iota A$ are **modally equivalent in A** if $\mathcal{L}_A x = \mathcal{L}_A y$. We denote the modal equivalence relation for A by $\sim_A$.

4.1.2 Patterns

In the following, we need a notion of patterns. The idea of patterns is due to Kaminski and Smolka [19].

Definition 14 (Pattern) A **pattern** is a set $\{\Box \rho_1 t_1, \ldots, \Box \rho_n t_n\}$ with $n \in \mathbb{N}$ of several $\Box$-expressions.

Definition 15 (Pattern of a Nominal) Let A be a branch. The pattern of a nominal $x \in \mathcal{N}_\iota A$ with respect to an atomic relation r is defined as follows.

$$\mathcal{P}^r_A x := \{\Box \rho t \mid \Box \rho t x \in A, \rho \in \{r, r^*\}\}$$

Note that the r-request of the labels of a nominal is exactly the r-request of the r-pattern of the nominal.

Proposition 9 Let A be a branch, r be an atomic relation, and $x \in \mathcal{N}_\iota A$. Then $\mathcal{R}^r(\mathcal{P}^r_A x) = \mathcal{R}^r(\mathcal{L}_A x)$.

Proof. It suffices to show for all $\rho \in \{r, r^*\}$ and for all t that $\Box \rho t x \in \mathcal{L}_A x \iff \Box \rho t x \in \mathcal{P}^r_A x$. This is immediate from Def. 15, Def. 12 and Def. 6. □

Definition 16 (Pattern Equivalence) Let A be a branch and $x, y \in \mathcal{N}_\iota A$. Then $x \sim^r_A y$ if $\mathcal{P}^r_A x = \mathcal{P}^r_A y$.

Definition 17 (Pattern Inclusion) Let A be a branch and $x, y \in \mathcal{N}_\iota A$. Then $x \subseteq^r_A y$ if $\mathcal{P}^r_A x \subseteq \mathcal{P}^r_A y$.

4.1.3 Relations

Branches encode a relational structure, and we introduce some notation to make this structure explicit, like we did it for interpretations in 3.1 before.

Definition 18 (*r*-Reachability, Reachability) Let A be a branch, and $x, y \in \mathcal{N}_\iota A$.

$$x \xrightarrow{r}_A y :\iff rxy \in A$$
$$x \longrightarrow_A y :\iff \exists r\colon x \xrightarrow{r}_A y$$

Definition 19 (*r*-Reachability Modulo Pattern Equivalence) Let A be a branch, $x, y \in \mathcal{N}_\iota A$ and r be a relation.

$$x \xrightarrow{\subseteq, r}_A y :\iff \exists x' \in \mathcal{N}_\iota A\colon x \subseteq^r_A x' \wedge x' \xrightarrow{r}_A y$$

The new categories from Def. 10 give rise to other kinds of relations. The α-relation models transition between nominals that both contain the same path variable and are linked by a relation r.

Definition 20 (α-Relation) Let A be a branch and $x, y \in \mathcal{N}_\iota A$ and $\alpha \in \mathcal{V}A$.

$$x \xrightarrow{\alpha}_A y \iff \alpha x, rxy, \alpha y \in A$$

We later establish that the r-transition in Def. 20 corresponds to the relation occurring the equation for α. Now we abstract from the path variables and relate nominals depending on the $\Diamond^*$-expression of the path variable.

Definition 21 ($\Diamond^*$-Relation) Let A be a branch and $x, y \in \mathcal{N}_\iota A$.

$$x \xrightarrow{\Diamond r^* t}_A y \iff \exists \alpha \in \mathcal{V}A\colon \alpha = \Diamond r^* t \in A \wedge x \xrightarrow{\alpha}_A y$$

Proposition 10 Let A be a branch and $\alpha = \Diamond r^* t \in A$. Then $\xrightarrow{\alpha}_A \subseteq \xrightarrow{\Diamond r^* t}_A$.

4.1.4 Paths

Relational structures give rise to the notion of paths.

Definition 22 (Path) Let S be a set and $\longrightarrow \in S \times S$. A **$\longrightarrow$-path** is a tuple $(x_0, \ldots, x_n)$ such that

- $\forall i \in [0, n]\colon x_i \in S$
- $\forall i \in [0, n-1]\colon x_i \longrightarrow x_{i+1}$

Definition 23 (Path Maximality) Let S be a set and $\longrightarrow \in S \times S$. A path $(x, \ldots, y)$ is **maximal**, if it cannot be extended: $\neg\exists z \in S\colon y \longrightarrow z$.

Definition 24 (Cyclic Paths) Let S be a set and $\longrightarrow \in S \times S$. A path $(x_0, \ldots, x_n)$ is **cyclic**, if there are $i, j \in [0, n]$ such that $i \neq j$ and $x_i = x_j$. A path is **acyclic**, if it is not cyclic.

Figure 4.1: Evidence Conditions

$$
\begin{array}{rll}
(\dot\neg p)x \in A \implies & px \notin A & \mathcal{E}_{\dot\neg} \\
(t_1 \dot\wedge t_2)x \in A \implies & t_1x \in A \;\wedge\; t_2x \in A & \mathcal{E}_{\dot\wedge} \\
(t_1 \dot\vee t_2)x \in A \implies & t_1x \in A \;\vee\; t_2x \in A & \mathcal{E}_{\dot\vee} \\
\Diamond rtx \in A \implies & \exists y \in \mathcal{N}_\iota A\colon x \xrightarrow{\subseteq,r}_A y \;\wedge\; ty \in A & \mathcal{E}_{\Diamond} \\
\Diamond r^*tx \in A \implies & \exists y \in \mathcal{N}_\iota A\colon x \xrightarrow{\subseteq,r}{}^*_A y \;\wedge\; ty \in A & \mathcal{E}_{\Diamond^*} \\
\Box rtx \in A \implies & \forall y \in \mathcal{N}_\iota A\colon x \xrightarrow{r}_A y \implies ty \in A & \mathcal{E}_{\Box} \\
\Box r^*tx \in A \implies & tx \in A \;\wedge\; \forall y \in \mathcal{N}_\iota A\colon x \xrightarrow{r}_A y \implies \Box r^*ty \in A & \mathcal{E}_{\Box^*}
\end{array}
$$

4.2 Model Existence

In this section, we establish a model existence theorem. We prove that every branch that satisfies a set of conditions called evidence conditions has a model. The formulation of the evidence conditions is based on the notion of patterns. Evidence conditions will be such that they do not take path variables into account.

4.2.1 Evidence Conditions

The evidence conditions are given in Figure 4.1. A branch that satisfies them is called **evident**. In fact, the conditions entail some stronger properties. This is captured in the following proposition.

Proposition 11 (Explicit Evidence) Let A be an evident set of K*-formulas. Then A additionally satisfies the following conditions.

$$
\begin{array}{rll}
\Box rtx \in A \implies & \forall y \in \mathcal{N}_\iota A\colon x \xrightarrow{\subseteq,r}_A y \implies ty \in A & \mathcal{E}'_{\Box} \\
\Box r^*tx \in A \implies & \forall y \in \mathcal{N}_\iota A\colon x \xrightarrow{\subseteq,r}{}^*_A y \implies ty \in A \;\wedge\; \Box r^*ty \in A & \mathcal{E}'_{\Box^*}
\end{array}
$$

Proof. Let A be an evident set of K*-formulas. For the proof of $\mathcal{E}'_{\Box}$, let $y \in \mathcal{N}_\iota A$.

$$
\begin{array}{lll}
& \Box rtx \in A \wedge x \xrightarrow{\subseteq,r}_A y & \\
\iff & \Box rtx \in A \wedge \exists x' \in \mathcal{N}_\iota A\colon x \subseteq^r_A x' \wedge x' \xrightarrow{r}_A y & \text{Def. 19} \\
\iff & \Box rtx \in A \wedge \exists x' \in \mathcal{N}_\iota A\colon \mathcal{P}^r_A x \subseteq \mathcal{P}^r_A x' \wedge x' \xrightarrow{r}_A y & \text{Def. 16} \\
\implies & \Box rtx' \in A \wedge x' \xrightarrow{r}_A y & \Box rt \in \mathcal{P}^r_A x \\
\implies & ty \in A & \mathcal{E}_{\Box}
\end{array}
$$

For the proof of $\mathcal{E}'_{\Box^*}$, let $\Box r^* tx \in A$, $y \in \mathcal{N}_\iota A$, and $x \xrightarrow{\subseteq,r}{}^{*}_A y$. Then $\exists n \in \mathbb{N}\colon x \xrightarrow{\subseteq,r}{}^{n}_A y$. We show $\Box r^* ty, ty \in A$ by induction on n.

Let $n = 0$. Then $x = y$. The claim follows with $\mathcal{E}_{\Box^*}$.

Let $n > 0$. Then $\exists z \in \mathcal{N}_\iota A\colon x \xrightarrow{\subseteq,r}{}^{n-1}_A z \wedge z \xrightarrow{\subseteq,r}{}^{1}_A y$. By Induction we get $\Box r^* tz, tz \in A$. Then $\exists z' \in \mathcal{N}_\iota A\colon z \sim^r_A z' \wedge z' \xrightarrow{r}_A y$ by Definition 19. $\Box r^* tz' \in A$ since $\mathcal{P}^r_A z' \subseteq \mathcal{P}^r_A z$. Thus $\Box r^* ty, ty \in A$ by $\mathcal{E}_{\Box^*}$. $\Box$

4.2.2 Model Existence Theorem

Let A be an evident branch. We now construct a satisfying modal interpretation $\mathcal{I}$.

$$\begin{aligned} \mathcal{I}I &:= \mathcal{N}_\iota A \\ \mathcal{I}x &:= x \\ \mathcal{I}rxy &:= x \xrightarrow{\subseteq,r}_A y && r \text{ atomic} \\ \mathcal{I}px &:= px \in A && p \text{ atomic} \end{aligned}$$

Theorem 2 Every (finite) evident set is (finitely) satisfiable.

Proof. We prove $tx \in A \implies \mathcal{I} \models tx$ by induction on the well-founded subterm relation of t. Case analysis.

Let $px \in A$.

$$\begin{aligned} px \in A &\iff \mathcal{I}px = 1 \\ &\iff \hat{\mathcal{I}}px = 1 \\ &\iff \mathcal{I} \models px \end{aligned}$$

Let $\dot\neg px \in A$.

$$\begin{aligned} \dot\neg px \in A &\implies px \notin A && \mathcal{E}_{\dot\neg} \\ &\iff \mathcal{I} \not\models px && \text{Case } px \in A \end{aligned}$$

Let $(t_1 \dot\wedge t_2)x \in A$.

$$\begin{aligned} (t_1 \dot\wedge t_2)x \in A &\implies t_1x, t_2x \in A && \mathcal{E}_{\dot\wedge} \\ &\implies \mathcal{I} \models t_1x \wedge \mathcal{I} \models t_2x && \text{Induction} \\ &\iff \mathcal{I} \models t_1x \wedge t_2x \\ &\iff \mathcal{I} \models (t_1 \dot\wedge t_2)x && \text{Def. } \dot\wedge \end{aligned}$$

Let $(t_1 \dot{\vee} t_2)x \in A$.

$$\begin{aligned}
(t_1 \dot{\vee} t_2)x \in A &\implies t_1x \in A \vee t_2x \in A && \mathcal{E}_{\dot{\vee}} \\
&\implies \mathcal{I} \models t_1x \vee \mathcal{I} \models t_2x && \text{Induction} \\
&\iff \mathcal{I} \models t_1x \vee t_2x && \\
&\iff \mathcal{I} \models (t_1 \dot{\vee} t_2)x && \text{Def. } \dot{\vee}
\end{aligned}$$

Let $\Diamond rtx \in A$.

$$\begin{aligned}
\Diamond rtx \in A &\implies \exists y \in \mathcal{N}_\iota A\colon x \xrightarrow{\subseteq, r}_A y \wedge ty \in A && \mathcal{E}_\Diamond, y \notin \mathcal{N}_\iota t \cup \{x\} \\
&\implies \exists y \in \mathcal{N}_\iota A\colon x \xrightarrow{\subseteq, r}_A y \wedge \mathcal{I} \models ty && \text{Induction} \\
&\iff \exists y \in \mathcal{N}_\iota A\colon \mathcal{I}rxy = 1 \wedge \mathcal{I} \models ty && \text{Def. } \mathcal{I} \\
&\iff \exists y \in \mathcal{N}_\iota A\colon \mathcal{I} \models rxy \wedge \mathcal{I} \models ty && \text{Def. } \models \\
&\iff \exists y \in \mathcal{N}_\iota A\colon \mathcal{I} \models rxy \wedge ty && \\
&\iff \exists y \in \mathcal{N}_\iota A\colon \mathcal{I} \models (\lambda y.rxy \wedge ty)y && \beta\text{-law} \\
&\iff \mathcal{I} \models \exists y.\ rxy \wedge ty && \text{Proposition 2} \\
&\iff \mathcal{I} \models \Diamond rtx && \text{Def. } \Diamond
\end{aligned}$$

Let $\Diamond r^*tx \in A$.

$$\begin{aligned}
\Diamond r^*tx \in A &\implies \exists y \in \mathcal{N}_\iota A\colon x \xrightarrow{\subseteq, r}{}^*_A y \wedge ty \in A && \mathcal{E}_{\Diamond^*}, y \notin \mathcal{N}_\iota t \cup \{x\} \\
&\implies \exists y \in \mathcal{N}_\iota A\colon x \xrightarrow{\subseteq, r}{}^*_A y \wedge \mathcal{I} \models ty && \text{Induction} \\
&\iff \exists y \in \mathcal{N}_\iota A\colon \mathcal{I}r^*xy = 1 \wedge \mathcal{I} \models ty && \text{Prop. 5} \\
&\iff \exists y \in \mathcal{N}_\iota A\colon \mathcal{I} \models r^*xy \wedge \mathcal{I} \models ty && \text{Def. } \models \\
&\iff \exists y \in \mathcal{N}_\iota A\colon \mathcal{I} \models r^*xy \wedge ty && \\
&\iff \exists y \in \mathcal{N}_\iota A\colon \mathcal{I} \models (\lambda y.r^*xy \wedge ty)y && \beta\text{-law} \\
&\iff \mathcal{I} \models \exists y.\ r^*xy \wedge ty && \text{Proposition 2} \\
&\iff \mathcal{I} \models \Diamond r^*tx && \text{Def. } \Diamond
\end{aligned}$$

Let $\Box rtx \in A$.

$$
\begin{aligned}
\Box rtx \in A &\implies \forall y \in \mathcal{N}_\iota A \colon x \xrightarrow{\subseteq,r}_A y \implies ty \in A && \mathcal{E}'_\Box, y \notin \mathcal{N}_\iota t \cup \{x\} \\
&\implies \forall y \in \mathcal{N}_\iota A \colon x \xrightarrow{\subseteq,r}_A y \implies \mathcal{I} \models ty && \text{Induction} \\
&\iff \forall y \in \mathcal{N}_\iota A \colon \mathcal{I}rxy = 1 \implies \mathcal{I} \models ty && \text{Def. } \mathcal{I} \\
&\iff \forall y \in \mathcal{N}_\iota A \colon \mathcal{I} \models rxy \implies \mathcal{I} \models ty && \text{Def. } \models \\
&\iff \forall y \in \mathcal{N}_\iota A \colon \mathcal{I} \models rxy \to ty && \\
&\iff \forall y \in \mathcal{N}_\iota A \colon \mathcal{I} \models (\lambda y.rxy \to ty)y && \beta\text{-law} \\
&\iff \mathcal{I} \models \forall y.\ rxy \to ty && \text{Proposition 2} \\
&\iff \mathcal{I} \models \Box rtx && \text{Def. } \Box
\end{aligned}
$$

Let $\Box r^*tx \in A$.

$$
\begin{aligned}
\Box r^*tx \in A &\implies \forall y \in \mathcal{N}_\iota A \colon x \xrightarrow{\subseteq,r}{}^*_A\, y \implies ty \in A && \mathcal{E}'_\Box, y \notin \mathcal{N}_\iota t \cup \{x\} \\
&\implies \forall y \in \mathcal{N}_\iota A \colon x \xrightarrow{\subseteq,r}{}^*_A\, y \implies \mathcal{I} \models ty && \text{Induction} \\
&\iff \forall y \in \mathcal{N}_\iota A \colon \mathcal{I}r^*xy = 1 \implies \mathcal{I} \models ty && \text{Prop. 5} \\
&\iff \forall y \in \mathcal{N}_\iota A \colon \mathcal{I} \models r^*xy \implies \mathcal{I} \models ty && \text{Def. } \models \\
&\iff \forall y \in \mathcal{N}_\iota A \colon \mathcal{I} \models r^*xy \to ty && \\
&\iff \forall y \in \mathcal{N}_\iota A \colon \mathcal{I} \models (\lambda y.r^*xy \to ty)y && \beta\text{-law} \\
&\iff \mathcal{I} \models \forall y.\ r^*xy \to ty && \text{Proposition 2} \\
&\iff \mathcal{I} \models \Box r^*tx && \text{Def. } \Box
\end{aligned}
$$

□

4.3 Rules

The schemata for the tableau rules are depicted in Figure 4.2. Note that we introduce a path variable for every $\Diamond^*$-formula. This is necessary to prove straightness of the system.

4.4 The Basic System $\mathcal{T}$

We impose progress conditions as usual on the basic system.

Definition 25 (Progress Conditions) Let $\langle A, A_1, \ldots, A_n \rangle \in \mathcal{T}$ and $i \in [1, n]$. Then

P1 $\langle A \rangle \notin \mathcal{T}$

P2 $A \subsetneq A_i$

Figure 4.2: Tableau Rules of $\mathcal{T}$

$$\mathcal{T}_{\dot\neg}\ \frac{(\dot\neg p)x, px}{} \qquad \mathcal{T}_{\dot\wedge}\ \frac{(t_1 \dot\wedge t_2)x}{t_1 x\,,\ t_2 x} \qquad \mathcal{T}_{\dot\vee}\ \frac{(t_1 \dot\vee t_2)x}{t_1 x \mid t_2 x}$$

$$\mathcal{T}_{\Box}\ \frac{\Box rtx, rxy}{ty} \qquad \mathcal{T}^{R}_{\Box^*}\ \frac{\Box r^* tx}{tx} \qquad \mathcal{T}^{T}_{\Box^*}\ \frac{\Box r^* tx, rxy}{\Box r^* ty}$$

$$\mathcal{T}_{\Diamond}\ \frac{\Diamond rux}{rxy\,,\ uy}\ y \notin \mathcal{N}_\iota A \qquad \mathcal{T}^{\alpha}_{\Diamond^*}\ \frac{\Diamond r^* tx}{\alpha x, \alpha = \Diamond r^* t}\ \alpha \notin \mathcal{V}A \qquad \mathcal{T}_{\Diamond^*}\ \frac{\alpha x, \alpha = \Diamond r^* t}{tx \mid \Diamond r \alpha x}$$

P3 $x \in \mathcal{N}_\iota A_i - \mathcal{N}_\iota A \implies \forall y \in \mathcal{N}_\iota A\colon A \subsetneq A^x_{iy}$

For our proofs it will be important that the labels of a nominal never change after a successor has been introduced. We thus prioritize the tableau rules to ensure this property.

Definition 26 (Propagated Nominal) Let A be a branch. We say $x \in \mathcal{N}_\iota A$ is **propagated** if no rule in $\mathcal{T}_{\dot\neg}, \mathcal{T}_{\dot\wedge}, \mathcal{T}_{\dot\vee}, \mathcal{T}_{\Box}, \mathcal{T}^{R}_{\Box^*}, \mathcal{T}^{T}_{\Box^*}, \mathcal{T}^{\alpha}_{\Diamond^*}, \mathcal{T}_{\Diamond^*}$ applies to formulas at x.

We augment $\mathcal{T}_{\Diamond}$ with a side condition as follows.

$$\mathcal{T}_{\Diamond}\ \frac{\Diamond rux}{rxy\,,\ uy}\ y \notin \mathcal{N}_\iota A$$

- x is propagated

We define the tableau system $\mathcal{T}$ as the largest subset of the union of the rules generated by the schemata $\mathcal{T}_{\dot\neg}, \mathcal{T}_{\dot\wedge}, \mathcal{T}_{\dot\vee}, \mathcal{T}_{\Box}, \mathcal{T}^{R}_{\Box^*}, \mathcal{T}^{T}_{\Box^*}, \mathcal{T}_{\Diamond}, \mathcal{T}^{\alpha}_{\Diamond^*}, \mathcal{T}_{\Diamond^*}$ that obeys the progress conditions and contains all closing rules.

Definition 27 (Propagated Branch) An admissible branch A is **propagated** if it satisfies all evidence conditions, except $\mathcal{E}_\Diamond$ and $\mathcal{E}_{\Diamond^*}$, and additionally meets the following requirements.

$$\begin{aligned} \Diamond r^* tx \in A &\implies \exists \alpha \in \mathcal{V}A\colon \alpha x, \alpha = \Diamond r^* t \in A && \mathcal{M}_{\Diamond^*} \\ \alpha x, \alpha = \Diamond r^* t \in A &\implies tx \in A \vee \Diamond r \alpha x \in A && \mathcal{M}_{\alpha} \end{aligned}$$

From Def. 26 one would expect a branch to be propagated if all of its nominals are propagated. The next proposition states that this coincides with our definition of a propagated branch in Def. 27.

Proposition 12 Let A be an admissible branch. If no rule from $\mathcal{T} - \mathcal{T}_\Diamond$ applies to A, then A is propagated.

Proof. By contradiction. Case analysis.

Let $\dot\neg px \in A$ for the proof of $\mathcal{E}_{\dot\neg}$. Assume $px \in A$. Then $\mathcal{T}_{\dot\neg}$ applies to A. Contradiction.

Let $(t_1 \dot\wedge t_2)x \in A$ for the proof of $\mathcal{E}_{\dot\wedge}$. Assume $t_1x \notin A$ or $t_2x \notin A$. Then $\mathcal{T}_{\dot\wedge}$ applies. Contradiction.

Let $(t_1 \dot\vee t_2)x \in A$ for the proof of $\mathcal{E}_{\dot\vee}$. Assume $t_1x \notin A$ and $t_2x \notin A$. Then $\mathcal{T}_{\dot\vee}$ applies. Contradiction.

Let $\Box rtx \in A$ for the proof of $\mathcal{E}_\Box$. Assume $\exists y \in \mathcal{N}_\iota A\colon rxy \in A \;\wedge\; ty \notin A$. Then $\mathcal{T}_\Box$ applies. Contradiction.

Let $\Box r^*tx \in A$ for the proof of $\mathcal{E}_{\Box^*}$. Assume $tx \notin A$. Then $\mathcal{T}^R_{\Box^*}$ applies. Contradiction. Assume $\exists y \in \mathcal{N}_\iota A\colon rxy \wedge \Box r^*ty \notin A$. Then $\mathcal{T}^T_{\Box^*}$ applies. Contradiction.

Let $\Diamond r^*tx \in A$ for the proof of $\mathcal{M}_{\Diamond^*}$. Assume $\forall\alpha \in \mathcal{V}A\colon \{\alpha x, \alpha = \Diamond r^*t\} \nsubseteq A$. Then $\mathcal{R}^\alpha_{\Diamond^*}$ is applicable by Progress Condition P3. Contradiction.

Let $\alpha x, \alpha = \Diamond r^*t \in A$ for the proof of $\mathcal{M}_\alpha$. Assume $tx \notin A$ and $\Diamond r\alpha x \notin A$. Then $\mathcal{T}_{\Diamond^*}$ is applicable. Contradiction. □

4.5 Admissibility

We formalize important invariants on branches as admissibility criteria.

Definition 28 A branch A is **admissible**, if

1. $(\mathcal{N}_\iota A, \longrightarrow_A)$ is a tree
2. $\forall\alpha \in \mathcal{V}A\colon \{x \mid \alpha x \in A\}$ is an $\xrightarrow{\alpha}_A$-path
3. $\forall\alpha \in \mathcal{V}A$ there is a unique formula $\alpha = \Diamond r^*t \in A$.
4. $\alpha x, rxy, \alpha y, \alpha = \Diamond r^*t \in A \implies \Diamond r\alpha x \in A$
5. $\alpha x, rxy, \alpha y \in A \implies \alpha = \Diamond r^*t \in A$
6. $\Diamond r\alpha x \in A \implies \alpha x \in A$
7. $rxy \in A \wedge \langle A, A_1, \ldots, A_n\rangle \in \mathcal{T} \implies \forall i \in [1,n]\colon \forall u\colon ux \in A \iff ux \in A_i$

These conditions are invariants the tableau system guarantees to preserve.

Proposition 13 The rules of $\mathcal{T}$ preserve admissibility.

Proof Sketch. 1. Holds, since $\mathcal{T}_\Diamond$ uses fresh names.

2. Holds, since a formula αx can only be added to a branch if either for the predecessor y we have $\alpha y \in A$ and for no other successor x' of y we have $\alpha x' \in A$, or $\alpha \notin \mathcal{V}A$.

3. Holds, since $\mathcal{T}^{\alpha}_{\Diamond^*}$ uses fresh names.

4. Holds, since there is no other way to add αy to a branch A if $\alpha \in \mathcal{V}A$ than the expansion of a formula $\Diamond r\alpha x$.

5. Holds, since there is no other way to add αy to a branch A if $\alpha \in \mathcal{V}A$ than the expansion of a formula $\Diamond r\alpha x$, which contains the right relation r by $\mathcal{T}_{\Diamond^*}$ and thus ensures $rxy \in A$.

6. Holds, since $\mathcal{T}_{\Diamond^*}$ can only add $\Diamond r\alpha x$ to a branch A if $\alpha x \in A$.

7. Holds, since $\mathcal{T}_\Diamond$ is only applicable to $\Diamond rux$ if no other rule applies to formulas at x. □

4.6 Soundness

Proposition 14 The tableau rules of $\mathcal{T}$ are sound.

Proof Sketch. We give modally valid formulas to justify the soundness of the tableau rules.

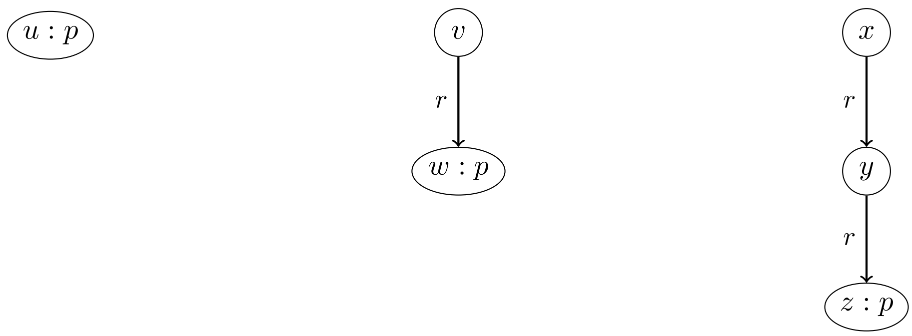
$$
\begin{array}{rll}
(p_1 \dot{\wedge} p_2)x \equiv & p_1x \wedge p_2x & \mathcal{T}_{\dot{\wedge}} \\
(p_1 \dot{\vee} p_2)x \equiv & p_1x \vee p_2x & \mathcal{T}_{\dot{\vee}} \\
\Box rpx \wedge rxy \rightarrow & py & \mathcal{T}_{\Box} \\
\Box r^*px \equiv & px \wedge \Box r(\Box r^*p)x & \mathcal{T}^{R}_{\Box^*}, \mathcal{T}^{T}_{\Box^*} \\
\Diamond rpx \rightarrow & \exists y.rxy \wedge py & \mathcal{T}_{\Diamond} \\
\Diamond r^*px \equiv & \alpha x \wedge \alpha = \Diamond r^*p & \mathcal{T}^{\alpha}_{\Diamond^*}, \mathcal{T}_{\Diamond^*} \\
\Diamond r^*px \equiv & px \vee \Diamond r(\Diamond r^*p)x & \mathcal{T}_{\Diamond^*}
\end{array}
$$
□

4.7 Straightness

We now define a strengthened soundness property called straightness. Soundness of a rule guarantees that if there is a model of the premise, there must be a model of at least one alternative among the conclusions. Straightness is the same property with respect to a certain set of branches. We formulate what it means for a branch to be straight through straightness conditions.

Before we start, we need some intuition. If our only goal is to satisfy $\Diamond r^*p$, we need an r-path to a state satisfying p. Such an r-path we call a **witness path**. Note that semantics of K* do not place any requirements on the length of the witness path. In Fig. 4.3, three models with witness paths of different length are given. The leftmost clearly features the shortest witness path. We have an interest in keeping witness paths in branches short.

Figure 4.3: Three witness paths of different length



Straight branches enjoy one nice property of particular importance: All their witness paths are short in some sense. At the end of this section, we will be able to prove that along a witness path on straight branches, a request never repeats before the witness occurs.

4.7.1 Witness Distances

We now devise a measure for the length of a witness path. Note that er assume min to yield ∞ for the empty set.

Definition 29 Given a modal interpretation $\mathcal{I}$, a state $a \in \mathcal{I}\iota$, a relation r and a modal expression t, the function $\delta^r_{\mathcal{I},t}\colon \mathcal{I}\iota \to \mathbb{N} \cup \{\infty\}$ yields the **witness distance**, which is the length of the shortest r-path in $\mathcal{I}$ starting at a and leading to a state satisfying t.

$$\delta^r_{\mathcal{I},t} a := \min\{n \in \mathbb{N} \mid \exists b \in \mathcal{I}\iota\colon a \xrightarrow{r}{}^{n}_{\mathcal{I}} b \wedge \hat{\mathcal{I}}tb = 1\}$$

Since a $\Diamond^*$-expression can be true in many states of a model, we are interested in the shortest witness path the model provides.

Definition 30 Given a modal interpretation $\mathcal{I}$, a set of modal expressions L, a relation r and a modal expression t, we define $\Delta^r_{\mathcal{I},t}\colon 2^{\mathcal{M}A} \to \mathbb{N} \cup \{\infty\}$, which yields the length of the shortest r-path in $\mathcal{I}$ from a state satisfying L to a state satisfying t.

$$\Delta^r_{\mathcal{I},t} L := \min\{\delta^r_{\mathcal{I},t} a \mid \mathcal{I}, a \models L\}$$

4.7.2 Selectors

We now parametrize the setup with a selector function. A selector function selects a subset of the labels of a nominal. We require all selectors to select at least enough to preserve the request of the nominal. Selectors make it possible to apply the same proof technique for two different blocking techniques.

Definition 31 Let A be a branch. A **selector** for A is a family of functions σ_A that is parametric in the relations occurring in A such that it holds $\forall x \in \mathcal{N}_\iota A \colon \mathcal{R}^r(\sigma^r_A x) = \mathcal{R}^r(\mathcal{L}_A x)$ for all relations r that occur in A.

4.7.3 Straight Branches

We now have all definitions needed to define the straightness conditions on branches.

Definition 32 Let A be a branch and $\mathcal{I}$ be an interpretation. $\mathcal{I}$ is **straight** for A if it satisfies the following conditions for all selectors σ_A:

S1 $\quad s \in A \implies \mathcal{I} \models s \quad$ if s is no edge

S2 $\quad rxy \in A \implies \mathcal{I}, \mathcal{I}y \models \mathcal{R}^r(\sigma^r_A x)$

O1 $\quad \alpha x, rxy, \alpha y, \alpha = \Diamond r^* t \in A \implies \delta^r_{\mathcal{I},t}(\mathcal{I}y) = \Delta^r_{\mathcal{I},t}(\mathcal{R}^r(\sigma^r_A x))$

O2 $\quad \alpha x, \alpha = \Diamond r^* t, \Diamond r \alpha x \in A \implies \mathcal{I} \not\models tx$

We say A is **straight** if there is an interpretation that is straight for A.

The crucial property of straight branches is that the witness distance always decreases along the corresponding $\Diamond^*$-relation.

Proposition 15 Let A be an admissible, straight branch, and $\mathcal{I}$ be straight for A. If $x \xrightarrow{\Diamond r^* t}_A y$ and $\mathcal{I} \not\models ty$, then $\Delta^r_{\mathcal{I},t}(\mathcal{R}^r(\sigma^r_A x)) > \Delta^r_{\mathcal{I},t}(\mathcal{R}^r(\sigma^r_A y))$ for all selectors σ^r_A.

Proof.

$$
\begin{array}{lll}
& x \xrightarrow{\Diamond r^* t}_A y & \\
\iff & \exists \alpha \in \mathcal{V}A \colon \alpha = \Diamond r^* t, \alpha x, rxy, \alpha y \in A & \text{Def. 21, Def. 20} \\
\implies & \exists \alpha \in \mathcal{V}A \colon \mathcal{I} \models \{\alpha = \Diamond r^* t, \alpha y\} & S1 \text{ for } \mathcal{I} \\
\implies & \exists n \in \mathbb{N} \colon \exists c \in \mathcal{I}\iota \colon (\mathcal{I}y) \xrightarrow{r}{}^{n}_{\mathcal{I}} c \wedge \hat{\mathcal{I}}tc = 1 & \text{Prop. 4} \\
\implies & \exists n \in \mathbb{N} \colon \exists b, c \in \mathcal{I}\iota \colon (\mathcal{I}y) \xrightarrow{r}{}^{1}_{\mathcal{I}} b \wedge b \xrightarrow{r}{}^{n-1}_{\mathcal{I}} c \wedge \hat{\mathcal{I}}tc = 1 & \mathcal{I} \not\models ty
\end{array}
$$

Now we fix the least $n \in \mathbb{N}$ that satisfies the equation above. We get $\delta^r_{\mathcal{I},t} b = \delta^r_{\mathcal{I},t}(\mathcal{I}y) - 1$ since n is chosen minimally. By Proposition 6 we have $\mathcal{I}, b \models \mathcal{R}^r(\mathcal{L}_A y)$. Now consider the following:

$$
\begin{array}{rll}
\Delta^r_{\mathcal{I},t}(\mathcal{R}^r(\sigma^r_A y)) & = \min\{\delta^r_{\mathcal{I},t} a \mid \mathcal{I}, a \models \mathcal{R}^r(\sigma^r_A y)\} & \text{Def. } \Delta^r_{\mathcal{I},A} \\
& \leq \delta^r_{\mathcal{I},t} b & \mathcal{I}, b \models \mathcal{R}^r(\sigma^r_A y), \text{Def. 31} \\
& < \delta^r_{\mathcal{I},t}(\mathcal{I}y) & \delta^r_{\mathcal{I},t} b = \delta^r_{\mathcal{I},t}(\mathcal{I}y) - 1 \\
& = \Delta^r_{\mathcal{I},t}(\mathcal{R}^r(\sigma^r_A x)) & \text{O1}
\end{array}
$$

□

Given a repetition of a selection along a $\Diamond^*$-path, the above proposition immediately yields a contradiction, since the witness distance only depends on the selection.

4.7.4 Straightness of $\mathcal{T}$

Now, we want to make sure that $\mathcal{T}$ indeed enjoys the straightness property.

Proposition 16 Let A be an admissible, straight branch and $\langle A, A_1, \ldots, A_n\rangle \in \mathcal{T}$. Then there is an $i \in [1, n]$ such that A_i is straight.

Proof. Since A is straight, there is a modal interpretation $\mathcal{I}$ that is straight for A. Let σ^r_A be a selector. Case analysis on the schema $\mathcal{R}$ of the applied rule.

Let $\langle A\rangle \in \mathcal{R}_{\dot\neg}$ with $px, \dot\neg px \in A$. We show by contradiction that this rule never becomes applicable. By S1 for A, $\mathcal{I} \models px \wedge \mathcal{I} \models \dot\neg px$, a contradiction.

Let $\langle A, A \cup \{t_1x\}, A \cup \{t_2x\}\rangle \in \mathcal{R}_{\dot\vee}$ with $(t_1 \dot\vee t_2)x \in A$. S1 for A yields $\mathcal{I} \models (t_1 \dot\vee t_2)x$, thus $\mathcal{I} \models t_1x \vee \mathcal{I} \models t_2x$. We can choose the branch $i \in \{1, 2\}$ such that Condition S1 holds for $A \cup \{t_i\}$. Condition O1 and Condition O2 hold, since the disjunction cannot contain a path variable by admissibility of A.

Let $\langle A, A \cup \{t_1x, t_2x\}\rangle \in \mathcal{R}_{\dot\wedge}$ with $(t_1 \dot\wedge t_2)x \in A$. By S1 for A we get $\mathcal{I} \models (t_1 \dot\wedge t_2)x$, thus $\mathcal{I} \models t_1x \wedge \mathcal{I} \models t_2x$, which ensures Condition S1. Condition O1 and Condition O2 hold, since the conjunction cannot contain a path variable by admissibility of A.

Let $\langle A, A \cup \{\alpha x, \alpha = \Diamond r^*t\}\rangle \in \mathcal{R}^{\alpha}_{\Diamond^*}$ with $\Diamond r^*tx \in A$. Since $\alpha \notin \mathcal{V}A$, we arrange things such that $\mathcal{I} \models \alpha = \Diamond r^*t$. Then we get $\mathcal{I} \models \alpha x$ since $\mathcal{I} \models \Diamond r^*tx$ by S1 for A. This yields Condition S1. Condition O1 and Condition O2 hold, since $\alpha \notin \mathcal{V}A$.

Let $\langle A, A \cup \{tx\}, A \cup \{\Diamond r\alpha x\}\rangle \in \mathcal{R}_{\Diamond^*}$ with $\alpha x, \alpha = \Diamond r^*t \in A$. By S1 for A we get $\mathcal{I} \models \Diamond r^*tx$, thus $\mathcal{I} \models tx$ or $\mathcal{I} \models \Diamond r\alpha x$. Case analysis.

- **Let** $\mathcal{I} \models tx$. We choose $A \cup \{tx\}$, so Condition S1 holds. Condition O1 and Condition O2 hold, since t does not contain path variables by admissibility of A.
- **Let** $\mathcal{I} \not\models tx$. We choose $A \cup \{\Diamond r\alpha x\}$. This ensures Condition S1 and Condition O2.

Let $\langle A, A \cup \{rxy, uy\}\rangle \in \mathcal{R}_{\Diamond}$ with $\Diamond rux \in A, y \notin \mathcal{N}_\iota A$. Since $\mathcal{I} \models \Diamond rux$ by S1 for A, there is $b \in \mathcal{I}\iota$ such that $(\mathcal{I}x) \xrightarrow{r}_{\mathcal{I}} b$ and $\hat{\mathcal{I}}ub = 1$ by Proposition 4. By Proposition 6, $\mathcal{I}, b \models \mathcal{R}^r(\mathcal{L}_Ax)$. Case analysis on the form of u.

- **Let** $u \in \mathcal{V}A$. Then $u = \Diamond r^*t \in A$. We choose $b' \in \mathcal{I}\iota$ such that $\delta^r_{\mathcal{I},t}b' = \Delta^r_{\mathcal{I},t}(\mathcal{R}^r(\sigma^r_Ax))$ and $\mathcal{I}, b' \models \mathcal{R}^r(\sigma^r_Ax)$. Note that since $\mathcal{R}^r(\mathcal{L}_Ax) = \mathcal{R}^r(\sigma^r_Ax)$, b itself is a possible candidate for such a b', thus it is sure that $\delta^r_{\mathcal{I},t}b' \in \mathbb{N}$. Setting $\mathcal{I}y := b'$ ensures Condition S1, since $\delta^r_{\mathcal{I},t}b' \in \mathbb{N}$ entails $\mathcal{I}, b' \models \Diamond r^*t$. We get Condition S2 and Condition O1 by the choice of b'. Condition O2 holds, since $y \notin \mathcal{N}_\iota A$, thus $\Diamond ruy \notin A$.

Let $u \notin \mathcal{V}A$. We set $\mathcal{I}y := b$ and get Condition S1. Condition S2 holds since $\mathcal{R}^r(\mathcal{L}_A x) = \mathcal{R}^r(\sigma^r_A x)$ by Definition 31. Condition O1 holds, since t contains no path variable since A is a branch and $y \notin \mathcal{N}_\iota A$. Condition O2 holds, since $y \notin \mathcal{N}_\iota A$, thus for all $\alpha \in \mathcal{V}A$, $\Diamond r\alpha y \notin A$.

Let $\langle A, A \cup \{ty\}\rangle \in \mathcal{R}_\Box$ with $\Box rtx, rxy \in A$. Since $\mathcal{I}, \mathcal{I}y \models \mathcal{R}^r(\sigma^r_A x)$ by S2 for A, and $t \in \mathcal{R}^r(\sigma^r_A x)$ by Def. 6, we get $\mathcal{I} \models ty$ by Prop. 6 and thus Condition S1. Condition O2 holds, since t contains no path variable by admissibility of A. We did not invalidate condition O1, since y has no successors by admissibility of A.

Let $\langle A, A \cup \{tx\}\rangle \in \mathcal{R}^R_{\Box *}$ with $\Box r^*tx \in A$. Since $\mathcal{I} \models \Box r^*tx$ by S1 for A, we get $\mathcal{I} \models ty$ from the tautologies in 2.2.5, thus we have Condition S1. Condition O2 holds, since t contains no path variable by admissibility of A. We did not invalidate condition O1, since x has no successors by admissibility of A.

Let $\langle A, A \cup \{\Box r^*ty\}\rangle \in \mathcal{R}^T_{\Box *}$ with $\Box r^*tx, rxy \in A$. Since $\mathcal{I}, \mathcal{I}y \models \mathcal{R}^r(\sigma^r_A x)$ by S2 for A, and $\Box r^*t \in \mathcal{R}^r(\sigma^r_A x)$ by Def. 6, we get $\mathcal{I} \models \Box r^*ty$ by Prop. 6 and thus Condition S1. Condition O2 holds, since t contains no path variable by admissibility of A. We did not invalidate condition O1, since y has no successors by admissibility of A. ☐

4.7.5 Path Variables and Straightness

In 4.3 we stated that path variables are necessary to prove straightness. We now demonstrate the point. The general problem is that we cannot formalize O1 without having path variables. Consider Example 1, which is a tableau derivation not using path variables.

Example 1 A hypothetical derivation without path variables.

$$
\begin{array}{r|l}
\multicolumn{2}{c}{\Diamond r^*px, \Box r(\Diamond r^*p)x, \Diamond rqx} \\
px & \Diamond r(\Diamond r^*p)x \\
 & rxy, qy \\
 & \Diamond r^*py
\end{array}
$$

It is not possible for the tableau to remember whether $\Diamond r^*py$ was added during expansion of $\Diamond r(\Diamond r^*p)x$ or $\Box r(\Diamond r^*p)x$. But this is important for the proof of straightness, because if the $\Box$-expansion added the formula, then the interpretation of y was already fixed and we cannot guarantee any condition like $O1$. Additionally, the expansion of $\Diamond r(\Diamond r^*p)x$ is not possible due to progress condition P2.

4.7.6 Straight Models

Up to now, we have not defined what exactly a **straight model** is. A model is straight, if it provides a direct successor with optimal witness distance for t for every state that satisfies $\Diamond r(\Diamond r^*t)$.

Definition 33 (Straight Model) Let $\mathcal{I}$ be a modal interpretation and L be a set of modal expressions. **$\mathcal{I}$ is straight for L**, if for all $a \in \mathcal{I}\iota$ and $\Diamond r(\Diamond r^* t) \in L$ such that $\mathcal{I}, a \models \Diamond r(\Diamond r^* t)$ there is a $b \in \mathcal{I}\iota$ such that $a \xrightarrow{r}_{\mathcal{I}} b$ and $\delta^{\mathcal{I}}_{\mathcal{I},t} b = \Delta^{\mathcal{I}}_{\mathcal{I},t}(\mathcal{R}^r(\mathcal{L}^L_{\mathcal{I}} a))$.

Note the correspondence between this definition and the condition O1 we required for straight branches.

Proposition 17 Let t be a modal expression and $\mathcal{I}$ be a canonical model for t. Then $\mathcal{I}$ is a straight model for $\mathcal{M}t$.

Proof. Let $a \in \mathcal{I}\iota$ and $\Diamond r(\Diamond r^* t) \in \mathcal{M}t$ such that $\mathcal{I}, a \models \Diamond r(\Diamond r^* t)$. Consider

$$\begin{aligned} & \Delta^{\mathcal{I}}_{\mathcal{I},t} a = \{\delta^{\mathcal{I}}_{\mathcal{I},t} b \mid \mathcal{I}, b \models \mathcal{R}^r a\} & \\ \Longrightarrow\ & \Delta^{\mathcal{I}}_{\mathcal{I},t} a = \{\delta^{\mathcal{I}}_{\mathcal{I},t} b \mid \mathcal{R}^r a \subseteq b\} & \text{Prop. 8}, \mathcal{R}^r a \subseteq \mathcal{C}t \\ \Longrightarrow\ & \Delta^{\mathcal{I}}_{\mathcal{I},t} a = \{\delta^{\mathcal{I}}_{\mathcal{I},t} b \mid \mathcal{I} \models rab\} & \text{Def. 9} \end{aligned}$$

The considered set is non-empty, since $\mathcal{I}$ is a model. □

5 Chain-based Control

In this chapter we equip the tableau system $\mathcal{T}$ with chain-based blocking and prove it complete. In Section 5.1 we define the blocking condition, and restrict $\mathcal{T}_\Diamond$ accordingly. In Section 5.2 we prove termination for the new system. In Section 5.3 we characterize maximal branches and develop conditions to identify evident ones. In Section 5.4 we finally obtain the completeness theorem.

5.1 Blocking

We use the established idea of chain-based blocking to ensure termination through restriction of rules applicability. This idea is due to Kripke [23]. The central concept of chain-based blocking is the notion of a repeating nominal.

Definition 34 Let A be a branch and $x, y \in \mathcal{N}_\iota A$. ***y* is repeating *x*** in A if $x \longrightarrow_A^+ y$ and $\mathcal{L}_A x = \mathcal{L}_A y$. A nominal $y \in \mathcal{N}_\iota A$ is repeating in A if it is repeating some $x \in \mathcal{N}_\iota A$.

The idea is, that for every repeating nominal, there already exists a nominal with exactly the same properties on the branch. Thus, instead of duplicating it, we just insert backward edges and obtain a model. Consider Example 2. $\Diamond rpz$ cannot be expanded, since it is repeating y. The encoded relational structure is not yet a model of the initial branch, we have to add the edge rzz, for example.

Example 2 A tableau derivation blocked by chain-based blocking for the initial branch $\{\Box r^*(\Diamond rp)x\}$.

$$\begin{array}{c} \Box r^*(\Diamond rp)x \\ \Diamond rpx \\ rxy, py \\ \Box r^*(\Diamond rp)y \\ \Diamond rpy \\ ryz, pz \\ \Box r^*(\Diamond rp)z \\ \Diamond rpz \end{array}$$

5.1.1 The Restricted System $\mathcal{T}_{chn}$

We are now ready to restrict $\mathcal{T}_\Diamond$ and $\mathcal{T}_{\Diamond^*}$ as follows.

$$\mathcal{R}_\Diamond \quad \frac{\Diamond r t x}{r x y\,,\; t y} \quad y \notin \mathcal{N}_\iota A$$

- x not repeating in A
- x is propagated

The system equipped with $\mathcal{R}_\Diamond$ in place of $\mathcal{T}_\Diamond$ is denoted by $\mathcal{T}_{chn}$.

5.2 Termination

We prove that every derivation in $\mathcal{T}_{chn}$ is finite. The proof precedes as follows. The crucial observation is that ExpStk(A) is finite and preserved under rule application. We then use that in admissible branches, $(\mathcal{N}_\iota A, \longrightarrow_A)$ is a tree. We bound breadth and depth of this tree, and get by König's lemma that the set of nominals ever introduced to a branch is finite. The difficulty here is to bound the number of $\Diamond$-formulas at a nominal, since we have to treat formulas $\Diamond r\alpha x$ as well. Finally we obtain a bound on the number of formulas ever added to a branch, which is sufficient for termination since our calculus is cumulative.

A key concept we use is the notion of a stock and a slack [20]. A stock is the set of all formulas that could possibly be introduced to the branch by a certain set of rules. The corresponding slack is the subset of the stock that has not been introduced so far. We then show that the size of the slack is strictly decreasing under application of rules from the set of considered rules.

5.2.1 Expressions

We define the **expression stock** ExpStk(A) of a branch A as the subset of proper modal expressions that occur, possibly as subexpressions, on A.

Definition 35 Let A be a branch.

$$\text{ExpStk}(A) := \{t \mid t \in \mathcal{M}A\}$$

Note that ExpStk(A) neither contains path variables, nor expressions that contain path variables. However, $\Diamond r^* t \in \text{ExpStk}(\{\alpha = \Diamond r^* t\})$.

Proposition 18 Let A be a finite branch. Then ExpStk(A) is finite.

It is easily checked that all rules preserve the expression stock.

Proposition 19 Let $\langle A, A_1, \ldots, A_n\rangle \in \mathcal{T}$. Then $\text{ExpStk}(A) = \text{ExpStk}(A_i)$ for all $i \in [1, n]$.

5.2.2 Variables

For $\mathcal{T}^{\alpha}_{\Diamond^*}$, we need another measure. We first define the **variable stock** VarStk(A), which is the set of all $\Diamond^*$-formulas possibly in A given $\mathcal{N}_\iota A$ and ExpStk(A). We then define the **variable slack** VarSlk(A), which contains all $\Diamond^*$-formulas from VarStk(A) for which no path variable has been introduced.

Definition 36 (Variable Stock, Variable Slack) Let A be a branch.

$$\begin{aligned} \mathrm{VarStk}(A) &:= \{\Diamond r^* tx \mid \Diamond r^* t \in \mathrm{ExpStk}(A), x \in \mathcal{N}_\iota A\} \\ \mathrm{VarSlk}(A) &:= \mathrm{VarStk}(A) - \{\Diamond r^* tx \mid \alpha x, \alpha = \Diamond r^* t \in A\} \end{aligned}$$

Proposition 20 Let A be a finite branch. Then VarStk(A) is finite.

Proposition 21 The rules of $\mathcal{T}$ preserve introduction of path variables.

Proposition 22 Let $\langle A, A_1, \ldots, A_n\rangle \in \mathcal{T} - \mathcal{T}^{\alpha}_{\Diamond^*}$. Then VarStk($A$) = VarStk($A_i$) for all $i \in [1, n]$.

Proposition 23 Let $\langle A, A_1\rangle \in \mathcal{T} \cap \mathcal{T}^{\alpha}_{\Diamond^*}$. Then $|\mathrm{VarSlk}(A)| > |\mathrm{VarSlk}(A_1)|$.

Proposition 24 The rules of $\mathcal{T}_{pat}$ are cumulative.

5.2.3 Termination Proof

Proposition 25 (Bound on Breadth) Let A be an admissible branch and $x \in \mathcal{N}_\iota A$. Then the number of direct successors of x is linearly bounded in the size of ExpStk(A) in $\mathcal{T}_{chn}$.

Proof. We have $\mathcal{L}_A x \subseteq \mathrm{ExpStk}(A)$. Together with Prop. 23 and Prop. 20 we get that the number of formulas αx in A is finite, thus the number of $\Diamond$-formulas at every nominal is finite by admissibility of A. By Progress Condition P3, a $\Diamond$-formula can only be expanded once, and by Prop. 24, expansion is preserved. Thus the number of direct successors of every nominal is finite. □

Proposition 26 (Bound on Depth) Let A be an admissible branch and $x \in \mathcal{N}_\iota A$. Then the number of predecessors of x is exponentially bounded in the size of ExpStk(A) in $\mathcal{T}_{chn}$.

Proof. Since $\mathcal{L}_A x \in 2^{\mathrm{ExpStk}(A)}$, there are only finitely many labels. By the Pigeon-hole principle, below a depth of at most $|2^{\mathrm{ExpStk}(A)}|$, a label must be repeating, thus below such a depth there is a repeating nominal by Def. 34. Since $\Diamond$-formulas cannot be expanded at repeating nominals, we have established a bound on the depth of x. □

Proposition 27 $\mathcal{T}_{chn}$ terminates.

Proof. For all admissible branches A, we have that $(\mathcal{N}_\iota A, \longrightarrow_A)$ is a tree by admissibility of A. From Prop. 25 and Prop. 26, it follows by König's lemma that this tree cannot grow infinitely, but its size is bounded, thus we have a bound on the number of nominals ever introduced during the expansion of A. Together with Prop. 23, Prop. 20, Prop. 18, and Prop. 19, the number of formulas ever added to the branch A is bounded. By Prop. 24, this is sufficient for termination. □

5.3 Maximal Tableaux

Consider Example 3. We have a maximal tableau with three open branches. Clearly, the rightmost branch is missing the witness for $\Diamond r^*px$, since none of the nominals satisfies p. Also, the rightmost branch is not straight, since x and y share the same request.

Example 3 A maximal tableau in $\mathcal{T}_{chn}$ for the initial branch $\{\Diamond r^*px\}$.

$$\begin{array}{c} \Diamond r^*px \\ \alpha = \Diamond r^*p, \alpha x \\ px \;\Big|\; \begin{array}{l} \Diamond r\alpha x \\ rxy, \alpha y \\ \;py \;\big|\; \Diamond r\alpha y \end{array} \end{array}$$

The question we have to answer is whether there will always be an evident branch in every maximal tableau for a satisfiable initial branch. The key is to prove evidence for straight branches. But this argument requires some preparation, so we first analyze the properties of the three kinds of maximal branches in $\mathcal{T}_{chn}$.

- Closed branches
- Maximal, open branches
- Maximal, open, and evident branches

5.3.1 Closed Branches

A branch A is **closed**, if $\langle A \rangle \in \mathcal{T}_{chn}$. A branch is **open**, if it is not closed. Closed branches contain a contradiction and are unsatisfiable.

5.3.2 Maximal, Open Branches

Maximal branches may fail to be evident because the tableau system cannot guarantee that every formula $\Diamond r^*tx$ is eventually expanded to ty at some nominal y. Consider the leftmost branch in Example 3, for example.

We now refine the modal equivalence relation to obtain $\trianglerighteq^\alpha_A$, which only relates a nominal x to other nominals, if x is labeled with the path variable α, and blocked by a predecessor that is also labeled with α. This means, we only consider pairs of nominals such that a path variable has been propagated from the second to the first. In Example 3, we would have $y \trianglerighteq^\alpha_A x$.

Definition 37 (α-Repetition Relation) Let A be an admissible branch and $x, y \in \mathcal{N}_\iota A$.

$$x \trianglerighteq_A^\alpha x' \iff \alpha = \Diamond r^* t, \alpha x, \alpha x' \in A \wedge x \sim_A x' \wedge x' \xrightarrow{r}{}_A^* x$$

Now we interpret the α-relation modulo the α-repetition relation. That means, we introduce edges to successors of modally equivalent predecessors that have the same path variable. In Example 3, we would have $y \xrightarrow{\trianglerighteq,\alpha}_A y$.

Definition 38 (α-Request Relation) Let A be an admissible branch and $x, y \in \mathcal{N}_\iota A$.

$$x \xrightarrow{\trianglerighteq,\alpha}_A y \iff \exists x' \in \mathcal{N}_\iota A \colon x \trianglerighteq_A^\alpha x' \wedge x' \xrightarrow{\alpha}_A y$$

Definition 39 (Maximality) An admissible set A is **maximal** if it is propagated, satisfies $\mathcal{E}_\Diamond$, and the following condition:

$$\Diamond r \alpha x, \alpha = \Diamond r^* t \in A \implies \exists y \in \mathcal{N}_\iota A \colon x \xrightarrow{\trianglerighteq,\alpha}_A y \qquad \mathcal{M}_\Diamond$$

Proposition 28 Let A be an admissible branch. If A is open and no rule from $\mathcal{T}_{chn}$ applies, then A is maximal.

Proof Sketch. A is propagated by Proposition Prop. 12. We show the missing two properties.

Let $\Diamond rtx \in A$ for the proof of $\mathcal{E}_\Diamond$. Assume $\forall y \in \mathcal{N}_\iota A \colon x \xrightarrow{\subseteq,r}_A y \implies ty \notin A$. Then x is not repeating in A, thus $\mathcal{R}_\Diamond$ is applicable. Contradiction

Let $\Diamond r \alpha x, \alpha = \Diamond r^* t \in A$ for the proof of $\mathcal{M}_\Diamond$. Assume it is not the case that $\exists y \in \mathcal{N}_\iota A \colon x \xrightarrow{\trianglerighteq,\alpha}_A y$. But then x is not repeating and thus $\mathcal{R}_\Diamond$ is applicable. Contradiction. □

Proposition 29 Let A be an admissible branch and $\alpha = \Diamond r^* t \in A$. Then $\xrightarrow{\trianglerighteq,\alpha}_A \subseteq \xrightarrow{\subseteq,r}_A$.

5.3.3 Maximal, Open and Evident Branches

Now we are interested in paths in the α-request relation.

Definition 40 (α-Request Path) Let A be an admissible branch and $\alpha x \in A$. A **request path** for α in A is a $\xrightarrow{\trianglerighteq,\alpha}_A$-path that starts at x and is $\xrightarrow{\trianglerighteq,\alpha}_A$-maximal.

The important property of acyclic α-request paths is that maximality implies the existence of the corresponding witness.

Lemma 2 Let A be a maximal, admissible branch, $\alpha = \Diamond r^* t \in A$, and $(x_0, \ldots, x_n)$ be a request path for $\alpha x \in A$ with $n \in \mathbb{N}$. If $(x_0, \ldots, x_n)$ is acyclic, then $t x_n \in A$.

Proof. We show that the assumption $tx_n \notin A$ leads to a contradiction. We analyze two cases.

Let $n = 0$.

$$
\begin{aligned}
& \alpha x \in A && \text{Def. 40} \\
\Longrightarrow\ & \Diamond r \alpha x \in A && \mathcal{M}_\alpha, tx \notin A \\
\Longrightarrow\ & \exists y \in \mathcal{N}_\iota A \colon x \xrightarrow{\trianglerighteq,\alpha}_A y && \mathcal{M}_\Diamond
\end{aligned}
$$

Let $n > 0$.

$$
\begin{aligned}
& x_{n-1} \xrightarrow{\trianglerighteq,\alpha}_A x_n && \text{Def. 50} \\
\Longleftrightarrow\ & \exists x' \in \mathcal{N}_\iota A \colon x_{n-1} \trianglerighteq_A^\alpha x' \wedge x' \xrightarrow{\alpha}_A x_n && \text{Def. 38} \\
\Longleftrightarrow\ & \exists x' \in \mathcal{N}_\iota A \colon x_{n-1} \trianglerighteq_A^\alpha x' \\
& \wedge\, \alpha x', r x' x_n, \alpha x_n \in A && \text{Def. 20} \\
\Longrightarrow\ & \alpha = \Diamond r^* t, \alpha x_n \in A && \text{Admissibility of } A \\
\Longrightarrow\ & \alpha = \Diamond r^* t, \Diamond r \alpha x_n \in A && \mathcal{M}_\alpha, tx_n \notin A \\
\Longrightarrow\ & \exists y \in \mathcal{N}_\iota A \colon x_n \xrightarrow{\trianglerighteq,\alpha}_A y && \mathcal{M}_\Diamond
\end{aligned}
$$

In both cases we have a contradiction, since $(x_0, \ldots, x_n)$ is $\xrightarrow{\trianglerighteq,\alpha}_A$-maximal by Definition 40. □

Having established this lemma, it is easy to formalize a condition to identify evident branches.

Definition 41 Let A be an admissible branch. A **loop** is a request path $(x_0, \ldots, x_n)$ such that $x_0 = x_n$ and $n > 0$.

Proposition 30 Let A be a maximal, admissible branch and $x \in \mathcal{N}_\iota A$. If A is loop-free, then A is evident.

Proof. Since A is maximal, it satisfies all evidence conditions except (possibly) $\mathcal{E}_{\Diamond^*}$. Let $\Diamond r^* tx \in A$. Since A is loop-free, no request path for in A is cyclic. By $\mathcal{M}_{\Diamond^*}$, $\alpha = \Diamond r^* t, \alpha x \in A$. By Lemma 2, the request path for αx leads to a state satisfying t. With Proposition 29, $\Diamond r^* tx$ is evident in A. □

5.4 Completeness of $\mathcal{T}_{chn}$

In this section we prove completeness. The proof idea is as follows. By Prop. 16, we know that every tableau for a satisfiable formula contains a straight branch. We lift Prop. 15 to the α-request relation and use this result to show that the α-request relation is loop-free on straight branches. Finally we get that a straight, maximal branch is evident by Prop. 30.

Proposition 31 Let A be a branch. Then $\mathcal{L}_A x$ is a selector.

Lemma 3 Let A be an admissible, straight branch, $\mathcal{I}$ be straight for A, and $\alpha = \Diamond r^* t \in A$. If $x \xrightarrow{\trianglerighteq,\alpha}_A y$ and $\mathcal{I} \not\models ty$, then $\Delta^r_{\mathcal{I},t}(\mathcal{L}_A x) > \Delta^r_{\mathcal{I},t}(\mathcal{L}_A y)$.

Proof.

$$
\begin{array}{lll}
& x \xrightarrow{\trianglerighteq,\alpha}_A y & \\
\iff & \exists x' \in \mathcal{N}_\iota A\colon x \trianglerighteq^\alpha_A x' \wedge x' \xrightarrow{\alpha}_A y & \text{Def. 48} \\
\implies & \exists x' \in \mathcal{N}_\iota A\colon x \sim_A x' \wedge x' \xrightarrow{\alpha}_A y & \text{Def. 47} \\
\implies & \exists x' \in \mathcal{N}_\iota A\colon x \sim_A x' \wedge \Delta^r_{\mathcal{I},t}(\mathcal{L}_A x') > \Delta^r_{\mathcal{I},t}(\mathcal{L}_A y) & \text{Prop. 15, Prop. 31} \\
\implies & \exists x' \in \mathcal{N}_\iota A\colon \mathcal{L}_A x = \mathcal{L}_A x' \wedge \Delta^r_{\mathcal{I},t}(\mathcal{L}_A x') > \Delta^r_{\mathcal{I},t}(\mathcal{L}_A y) & \text{Def. 16} \\
\implies & \Delta^r_{\mathcal{I},t}(\mathcal{L}_A x) > \Delta^r_{\mathcal{I},t}(\mathcal{L}_A y) & \square
\end{array}
$$

Lemma 4 Let A be an admissible, straight branch, $\mathcal{I}$ be a modal interpretation that is straight for A, $x, y \in \mathcal{N}_\iota A$, and $\alpha = \Diamond r^* t \in A$. If $x \xrightarrow{\trianglerighteq,\alpha}_A y$, then $\mathcal{I} \not\models tx$.

Proof.

$$
\begin{array}{lll}
& x \xrightarrow{\trianglerighteq,\alpha}_A y & \\
\iff & \exists x' \in \mathcal{N}_\iota A\colon x \trianglerighteq^\alpha_A x' \wedge x' \xrightarrow{\alpha}_A y & \text{Def. 38} \\
\iff & \exists x' \in \mathcal{N}_\iota A\colon \alpha = \Diamond r^* t, \alpha x, \alpha x' \in A \wedge x \sim_A x' \wedge x' \xrightarrow{r}{}^*_A x & \\
& \wedge\, x' \xrightarrow{\alpha}_A y & \text{Def. 38} \\
\implies & \alpha = \Diamond r^* t, \alpha x, \Diamond r \alpha x \in A & A \text{ admissible} \\
\implies & \mathcal{I} \not\models tx & \text{O2 for } A \quad \square
\end{array}
$$

Lemma 5 Let A be a maximal branch. If A is straight, then A is loop-free.

Proof. Since A is straight, there is an interpretation $\mathcal{I}$ that is straight for A. Assume $(x_0, \ldots, x_n)$ is a loop in A for contradiction. By Lemma 4, $\mathcal{I} \not\models tx_i$ for all $i \in [0, n-1]$. By Definition 41 it holds $x_0 = x_n$. Thus, $\mathcal{I} \not\models tx_n$. Lemma 3 yields $\Delta^r_{\mathcal{I},t}(\mathcal{L}_A x_0) > \Delta^r_{\mathcal{I},t}(\mathcal{L}_A x_n)$. But $\Delta^r_{\mathcal{I},t}(\mathcal{L}_A x_0) = \Delta^r_{\mathcal{I},t}(\mathcal{L}_A x_n)$, since $\Delta^r_{\mathcal{I},t}$ and $\mathcal{L}_A$ are functions. Contradiction. $\square$

Theorem 3 (Completeness) Every maximal $\mathcal{T}_{pat}$-tableau for an satisfiable set of K*-expressions contains an evident branch.

Proof. Every satisfiable set of K*-expressions L there is a straight initial branch, namely $\{tx_0 \mid t \in L\}$. Since L by construction contains no edges and no path variables, any modal interpretation satisfying L is straight for L. A maximal tableau for L contains a maximal branch A which is straight by Prop. 16. By Lem. 5, A is loop-free. With Prop. 30, A is evident. $\square$

6 Pattern-based Control

In this section, we apply pattern-based blocking to $\mathcal{T}$ and obtain a complete and terminating system. In Section 6.1, we explain the blocking technique and apply it to $\mathcal{T}$ to obtain the system $\mathcal{T}_{pat}$. In Section 6.2, we prove termination by embedding a suitable size measure into a lexical termination ordering. In Section 6.3, we characterize maximal branches and give conditions to identify evident branches. Finally in Section 6.4, we are able to prove completeness for $\mathcal{T}_{pat}$.

6.1 Blocking

We use pattern-based blocking which has first appeared in [19, 20]. The idea is to forbid diamond expansion if the diamond expression has already been expanded at another nominal with a *similar* pattern. Let us make this precise.

6.1.1 $\Diamond$-Patterns

Definition 42 ($\Diamond$-Pattern) A **$\Diamond$-pattern** is a set $\{\Diamond\rho t\} \cup \{\Box\rho_1 t_1, \ldots, \Box\rho_n t_n\}$ with $n \in \mathbb{N}$, i.e. a set consisting of one $\Diamond$-expression and arbitrarily many $\Box$-expressions. We may call a $\Diamond$-pattern a $\Diamond^*$-pattern to indicate that ρ is not an atomic relation.

Definition 43 (Pattern of a $\Diamond$-Formula) Let A be a branch. The **$\Diamond$-pattern of a formula** $\Diamond rux \in A$ denoted by $\mathcal{P}^r_A(\Diamond rux)$ is defined according to the following equations:

$$\mathcal{P}^r_A(\Diamond rtx) := \{\Diamond rt\} \cup \mathcal{P}^r_A x$$
$$\mathcal{P}^r_A(\Diamond r\alpha x) := \{\Diamond r^* t \mid \alpha = \Diamond r^* t \in A\} \cup \mathcal{P}^r_A x$$

Note that the definition of $\mathcal{P}^r_A$ could potentially allow more than one $\Diamond$-expression in a $\Diamond$-pattern. This is not the case for the branches we consider.

Proposition 32 Let A be an admissible branch and $\Diamond r\alpha x \in A$. Then $\mathcal{P}^r_A(\Diamond r\alpha x)$ is a $\Diamond^*$-pattern.

Proof. Follows from admissibility of A, which ensures that defining equations for path variables are unique. □

6.1.2 Realization of $\Diamond$-Patterns

We now define what it means for a pattern to be realized. The condition depends on the relation ρ of the diamond expression in the pattern. If ρ is an atomic relation, we use the notion of realization as it appears in [20]. If the relation is not basic, the pattern belongs to a formula of the form $\Diamond r\alpha x$ where $\alpha = \Diamond r^* t$. In this case we must be careful to allow expansion long enough for a witness to be produced.

Definition 44 ($\Diamond$-Pattern Realization) Let A be a branch and $x \in \mathcal{N}_\iota A$.

- $\{\Diamond rt\} \cup \mathcal{P}^r_A x$ is **realized** in A, if there is $y \in \mathcal{N}_\iota A$ such that $x \xrightarrow{\subseteq, r}_A y$, and $ty \in A$.
- $\{\Diamond r^* t\} \cup \mathcal{P}^r_A x$ is **realized** in A, if there is $x', y \in \mathcal{N}_\iota A$ such that $\mathcal{P}^r_A x' = \mathcal{P}^r_A x$ and $x' \xrightarrow{\Diamond r^* t}_A y$.

Note that for admissible branches A we have $\Diamond r\alpha x \in A$ in case of $\Diamond^*$-patterns.

6.1.3 The Restricted System $\mathcal{T}_{pat}$

We are now ready to restrict the diamond rule $\mathcal{T}_\Diamond$ as follows.

$$\mathcal{R}_\Diamond \quad \frac{\Diamond rux}{rxy\,,\; uy} \quad y \notin \mathcal{N}_\iota A$$

- $\mathcal{P}^r_A(\Diamond rux)$ not realized in A
- x is propagated

The system equipped with $\mathcal{R}_\Diamond$ in place of $\mathcal{T}_\Diamond$ is denoted by $\mathcal{T}_{pat}$.

6.1.4 Remark: Pattern Inclusion

For proper $\Diamond$-patterns we use pattern inclusion in the realization condition. A $\Diamond$-pattern is realized if there is an expanded $\Diamond$-formula for a pattern it is contained in, i.e. the pattern of the expanded formula contains more $\Box$-expressions. It turns out that pattern inclusion does not work for $\Diamond^*$-patterns. We show a maximal tableau which is by no means contains an evident branch as counterexample.

Example 4 A maximal derivation in a hypothetical system for a satisfiable initial branch without an evident branch.

$$\begin{array}{c}
\Diamond r^* px, \Box r(\dot\neg p)x, \dot\neg px \\
\alpha = \Diamond r^* p, \alpha x \\
\begin{array}{r|l}
px & \Diamond r\alpha x \\
 & rxy, \alpha y \\
 & \dot\neg py \\
 & \begin{array}{c|c} py & \Diamond r\alpha y \end{array}
\end{array}
\end{array}$$

The formula $\Diamond r\alpha y$ in the tableau from Example 4 cannot be be expanded since its pattern $\{\Diamond r^* p\}$ is a subset of the pattern $\{\Diamond r^* p, \Box r(\dot\neg p)\}$ which is realized at x. Although the initial branch is satisfiable, the tableaux contains no evident branch since the only open branch is missing the witness for the formula $\Diamond r^* px$. We have taken care of this problem by formulating separate realization conditions for $\Diamond^*$-patterns, such that expansion of formulas of the form $\Diamond r\alpha x$ is only blocked on the basis of pattern equality.

6.2 Termination

We now prove that every derivation in $\mathcal{T}_{pat}$ is finite. We use the result from Section 5.2 stating that no new modal expressions are introduced by $\mathcal{T}_{pat}$, which allows us to bound the number of patterns on a branch. The termination proof is an embedding of the size measures into a lexical termination ordering.

$\mathcal{T}_{pat}$ contains two generative rules, $\mathcal{R}_\Diamond$ and $\mathcal{T}_{\Diamond^*}$. To measure their progress, we will use the notion of a stock and a slack [20] as described in 5.2.

6.2.1 Patterns

We will now take care of the generative rules $\mathcal{R}_\Diamond$ and $\mathcal{T}^{\alpha}_{\Diamond^*}$ that introduce new names. We first take care of $\mathcal{R}_\Diamond$, and define the **pattern stock** $\mathrm{PatStk}(A)$.

Definition 45 (Pattern Stock, Pattern Slack) Let A be a branch.

$$\begin{aligned}\mathrm{PatStk}(A) &:= 2^{\mathrm{ExpStk}(A)}\\ \mathrm{PatSlk}(A) &:= \mathrm{PatStk}(A) - \{\mathcal{P}^r_A(\Diamond rux) \mid \Diamond rux \in A, \mathcal{P}^r_A(\Diamond rux) \text{ realized}\}\end{aligned}$$

Proposition 33 Let A be a finite branch. Then $\mathrm{PatStk}(A)$ is finite.

Proposition 34 The rules of $\mathcal{T}_{pat}$ preserve realization of $\Diamond$-patterns.

6.2.2 Formulas

We are ready to define the **formula stock** $\mathrm{ForStk}(A)$ for a branch A to be the set of all formulas possibly on the branch given $\mathcal{N}_\iota A$, $\mathcal{V}A$ and $\mathrm{ExpStk}(A)$. We then define the **formula slack** $\mathrm{ForSlk}(A)$, which contains the formulas from $\mathrm{ForStk}(A)$ that have not been introduced to the branch A.

Definition 46 (Formula Stock, Formula Slack) Let A be a branch.

$$\begin{aligned}\mathrm{ForStk}(A) &:= \{tx \mid t \in \mathrm{ExpStk}(A), x \in \mathcal{N}_\iota A\}\\ &\quad\cup \{\alpha x, \Diamond r\alpha x, \alpha = \Diamond r^* t \mid \Diamond r^* t \in \mathrm{ExpStk}(A), x \in \mathcal{N}_\iota A, \alpha \in \mathcal{V}A\}\\ \mathrm{ForSlk}(A) &:= \mathrm{ForStk}(A) - A\end{aligned}$$

Proposition 35 Let A be a finite branch. Then $\mathrm{ForStk}(A)$ is finite.

6.2.3 Termination Proof

Proposition 36 $\mathcal{T}_{pat}$ terminates.

Proof. Let A be a branch. Consider

- the size of the pattern slack of A, $|\text{PatSlk}(A)|$
- the size of the variable slack of A, $|\text{VarSlk}(A)|$
- the size of the formula slack of A, $|\text{ForSlk}(A)|$

The corresponding lexical ordering is a termination order for $\mathcal{T}_{pat}$, i.e. every application of a rule from $\mathcal{T}_{pat}$ to a branch A decreases the lexical product of the three size measures given above. Case analysis on the schema of the applied rule. If not stated otherwise, the size of PatSlk is preserved by Prop. 19 and 34, and the size of VarSlk is preserved by Prop. 21 and 22.

Let $\langle A\rangle \in \mathcal{T}_{\dot\neg}$ with $px, \dot\neg px \in A$. Nothing to show.

Let $\langle A, A \cup \{t_1x\}, A \cup \{t_2x\}\rangle \in \mathcal{T}_{\dot\vee}$ with $(t_1 \dot\vee t_2)x \in A$. Let $i \in \{1,2\}$. By Progress Condition P2, $t_ix \notin A$. Since $(t_1 \dot\vee t_2)x \in A$, we have $t_ix \in \text{ForStk}(A)$. Thus $|\text{ForSlk}(A \cup \{t_ix\})| < |\text{ForSlk}(A)|$.

Let $\langle A, A \cup \{t_1x, t_2x\}\rangle \in \mathcal{T}_{\dot\wedge}$ with $(t_1 \dot\wedge t_2)x \in A$. By Progress Condition P2, $t_ix \notin A$ for some $i \in \{1,2\}$. Since $(t_1 \dot\wedge t_2)x \in A$, we have $t_1x, t_2x \in \text{ForStk}(A)$. Thus $|\text{ForSlk}(A \cup \{t_1x, t_2x\})| < |\text{ForSlk}(A)|$.

Let $\langle A, A \cup \{\alpha x, \alpha = \Diamond r^*t\}\rangle \in \mathcal{T}^{\alpha}_{\Diamond^*}$ with $\Diamond r^*tx \in A$. Note that $\Diamond r^*tx \in \text{VarStk}(A)$. With Progress Condition P3 it follows $|\text{VarSlk}(A \cup \{\alpha x, \alpha = \Diamond r^*t\})| < |\text{VarSlk}(A)|$, since for every path variable $\beta \in \mathcal{V}A$ it holds $A \subsetneq (A \cup \{\alpha x, \alpha = \Diamond r^*t\})^{\alpha}_{\beta}$.

Let $\langle A, A \cup \{tx\}, A \cup \{\Diamond r\alpha x\}\rangle \in \mathcal{T}_{\Diamond^*}$ with $\alpha x, \alpha = \Diamond r^*t \in A$. Note that $t, \Diamond r^*t \in \text{ExpStk}(A)$, since $t, \Diamond r^*t \in \text{SubExps}(\{\alpha = \Diamond r^*t\})$. Since $tx \notin A$ by Progress Condition P2, we have $|\text{ForSlk}(A \cup \{tx\})| < |\text{ForSlk}(A)|$. Since further $\Diamond r\alpha x \notin \mathcal{V}A$, by Progress Condition P2, we have $|\text{ForSlk}(A \cup \{\Diamond r\alpha x\})| < |\text{ForSlk}(A)|$.

Let $\langle A, A \cup \{rxy, uy\}\rangle \in \mathcal{R}_{\Diamond}$ with $\Diamond rux \in A, y \notin \mathcal{N}_\iota A$. $|\text{PatSlk}(A \cup \{rxy, uy\})| < |\text{PatSlk}(A)|$ since $\mathcal{P}^r_A(\Diamond rux)$ has not been realized by blocking conditions of $\mathcal{R}_{\Diamond}$.

Let $\langle A, A \cup \{ty\}\rangle \in \mathcal{T}_{\Box}$ with $\Box rtx, rxy \in A$. By Progress Condition P2, $ty \notin A$. Since $\Box rtx, rxy \in A$, we have $ty \in \text{ForStk}(A)$. Thus $|\text{ForSlk}(A \cup \{ty\})| < |\text{ForSlk}(A)|$.

Let $\langle A, A \cup \{tx\}\rangle \in \mathcal{T}^R_{\Box^*}$ with $\Box r^*tx \in A$. By Progress Condition P2, $tx \notin A$. Since $\Box r^*tx \in A$, we have $tx \in \text{ForStk}(A)$. Thus $|\text{ForSlk}(A \cup \{tx\})| < |\text{ForSlk}(A)|$.

Let $\langle A, A \cup \{\Box r^*ty\}\rangle \in \mathcal{T}^T_{\Box^*}$ with $\Box r^*tx, rxy \in A$. By Progress Condition P2, $\Box r^*ty \notin A$. Since $\Box r^*tx, rxy \in A$, we have $\Box r^*ty \in \text{ForStk}(A)$. Thus we get that $|\text{ForSlk}(A \cup \{\Box r^*ty\})| < |\text{ForSlk}(A)|$. □

Next we will characterize maximal tableau branches.

6.3 Maximal Tableaux

In this section we investigate the properties of maximal tableau branches. Unlike usual systems, $\mathcal{T}_{pat}$ has three kinds of open branches:

- Closed branches
- Maximal, open branches
- Maximal, open, evident branches

6.3.1 Closed Branches

A branch A is **closed**, if $\langle A \rangle \in \mathcal{T}_{pat}$. A branch is **open**, if it is not closed. Note that open does not mean satisfiable. It merely means that we do not have an explicit inconsistency on the branch.

6.3.2 Maximal, Open Branches

Maximal branches may fail to be evident because the tableau system cannot guarantee that every formula $\Diamond r^*tx$ is eventually expanded to ty at some nominal y on every branch. Thus, maximal, open branches can be unsatisfiable. Consider the following example.

Example 5 A derivation in $\mathcal{T}_{pat}$ for the initial branch $\{\Diamond r^*px, \Box r^*(\dot\neg p)x\}$.

$$\begin{array}{c} \Diamond r^*px, \Box r^*(\dot\neg p)x \\ \alpha = \Diamond r^*p, \alpha x \\ \dot\neg px \\ \begin{array}{r|l} px & \Diamond r\alpha x \\ & rxy, \alpha y \\ & \Box r^*(\dot\neg p)y \\ & \dot\neg py \\ & \begin{array}{r|l} py & \Diamond r\alpha y \end{array} \end{array} \end{array}$$

Note that the rightmost branch is open. This example makes it clear that we need an argument to discard certain branches that contain no immediate contradiction.

We are now interested in the process of expansion of $\Diamond^*$-formulas. Thus we investigate how a formula αx is treated. Given αx, the tableau can either produce the witness, or request a successor that satisfies α. We now define a relation that relates a nominal x to a nominal y if either y has been produced to satisfy the request of x for a successor satisfying α, or y bears another path variable β for the same $\Diamond^*$-expression and was introduced to satisfy β. We call this relation $\Diamond^*$-request relation.

First, we restrict pattern equivalence such that a nominal x is only related to other nominals if there is a request for a successor for some path variable at x.

Definition 47 (Pattern Relation with Diamond Presence) Let A be a branch, and $x, x' \in \mathcal{N}_\iota A$.

$$x \trianglerighteq_A^{\Diamond r^*t} x' \iff \exists \alpha \in \mathcal{V}A\colon \alpha = \Diamond r^*t, \Diamond r\alpha x \in A \wedge x \sim_A^r x'$$

We now construct the $\Diamond^*$-request relation. This relation allows us to track the $\Diamond^*$-requests in our branch.

Definition 48 ($\Diamond^*$-Request Relation) Let A be a branch and $x, x' \in \mathcal{N}_\iota A$.

$$x \xrightarrow{\trianglerighteq, \Diamond r^* t}_A y \iff \exists x' \in \mathcal{N}_\iota A \colon x \trianglerighteq_A^{\Diamond r^* t} x' \land x' \xrightarrow{\Diamond r^* t}_A y$$

The $\Diamond^*$-request relation is a refinement of the corresponding reachability relation (Def. 19).

Proposition 37 Let A be an admissible branch and $\alpha \in \mathcal{V}A$ such that $\alpha = \Diamond r^* t \in A$. Then $\xrightarrow{\trianglerighteq, \Diamond r^* t}_A \subseteq \xrightarrow{\subseteq, r}_A$.

We now give maximality conditions. Note that we use the $\Diamond^*$-request relation for $\mathcal{M}_\Diamond$.

Definition 49 (Maximality) An admissible set A is **maximal** if it is propagated, satisfies $\mathcal{E}_\Diamond$ and the following condition:

$$\Diamond r \alpha x, \alpha = \Diamond r^* t \in A \implies \exists y \in \mathcal{N}_\iota A \colon x \xrightarrow{\trianglerighteq, \Diamond r^* t}_A y \qquad \mathcal{M}_\Diamond$$

Proposition 38 Let A be an admissible branch. If A is open and no rule from $\mathcal{T}_{pat}$ applies, then A is maximal.

Proof Sketch. By Prop. 12, A is propagated. We show the missing two properties:

Let $\Diamond rtx \in A$ for the proof of $\mathcal{E}_\Diamond$. Assume $\forall y \in \mathcal{N}_\iota A \colon x \xrightarrow{\subseteq, r}_A y \implies ty \notin A$. Then $\mathcal{P}_A^r(\Diamond rtx)$ is not realized in A, thus $\mathcal{R}_\Diamond$ is applicable. Contradiction

Let $\Diamond r \alpha x, \alpha = \Diamond r^* t \in A$ for the proof of $\mathcal{M}_\Diamond$. Assume it is not the case that $\exists y \in \mathcal{N}_\iota A \colon x \xrightarrow{\trianglerighteq, \Diamond r^* t}_A y$.

$$\begin{aligned} & \neg \exists y \in \mathcal{N}_\iota A \colon x \xrightarrow{\trianglerighteq, \Diamond r^* t}_A y \\ \iff & \neg \exists y \in \mathcal{N}_\iota A \colon \exists x' \in \mathcal{N}_\iota A \colon x \trianglerighteq_A^{\Diamond r^* t} x' \land x' \xrightarrow{\Diamond r^* t}_A y \\ \iff & \neg \exists y \in \mathcal{N}_\iota A \colon \exists x' \in \mathcal{N}_\iota A \colon \exists \alpha \in \mathcal{V}A \colon \alpha = \Diamond r^* t, \Diamond r \alpha x \in A \land x \sim_A^r x' \\ & \land x' \xrightarrow{\Diamond r^* t}_A y \end{aligned}$$

Since $\Diamond r \alpha x, \alpha = \Diamond r^* t \in A$ by assumption, the $\Diamond^*$-pattern $\mathcal{P}_A^r(\Diamond r \alpha x)$ is not realized and thus $\mathcal{R}_\Diamond$ is applicable. Contradiction. □

6.3.3 Maximal, Open, and Evident Branches

In this section we establish a condition on maximal branches that qualifies them as evident. We first give some intuition for two of the maximality conditions. Let us assume we have a branch A such that $\alpha = \Diamond r^* t, \alpha x \in A$.

$\mathcal{M}_\alpha$ *If tx was not added to A by $\mathcal{T}^\alpha_{\Diamond^*}$, then $\Diamond r\alpha x$ was added.*

$\mathcal{M}_\Diamond$ *If $\Diamond r\alpha x$ is on the branch, then there is a successor that satisfies α.*

Now assume tx has never been added in the course of the expansion of αx. By intuition $\mathcal{M}_\alpha$, for each nominal y with αy in the branch, but not ty, it was $\Diamond r\alpha y$ added. And by $\mathcal{M}_\Diamond$, for every such $\Diamond$-formula, there is a successor satisfying α. From the termination proof we know that the maximal branch contains only finitely many nominals. Thus we must have some sort of loop on the branch. We will use the existence of loops to identify non-evident branches.

The situation is complicated by the precise definition of the phrase *a successor that satisfies* α. First, successor is interpreted with respect to the $\xrightarrow{\subseteq, r}_A$-relation. And second, satisfying α may mean satisfying another path variable β such that $\beta = \Diamond r^* t \in A$. We have already defined a suitable formulation of the phrase *a successor that satisfies* α, namely the $\Diamond^*$-request relation.

Definition 50 ($\Diamond$*-Request Path) Let A be an admissible branch and $\Diamond r^* tx \in A$. A **$\Diamond$*-request path** for $\Diamond r^* tx$ in A is a $\xrightarrow{\trianglerighteq, \Diamond r^* t}_A$-path that starts at x and is $\xrightarrow{\trianglerighteq, \Diamond r^* t}_A$-maximal.

The important property of every $\Diamond^*$-request path is, that if it is maximal and not cyclic, we can prove the existence of the witness. This exactly matches our intuition for $\mathcal{M}_\alpha$.

Lemma 6 Let A be a maximal, admissible branch, and $(x_0, \dots, x_n)$ be a $\Diamond^*$-request path for $\Diamond r^* tx \in A$ with $n \in \mathbb{N}$. If $(x_0, \dots, x_n)$ is acyclic, then $\Diamond r^* tx$ is evident in A.

Proof. If $tx_n \in A$, then $\Diamond r^* tx$ is evident in A by Prop. 37. We show that the assumption $tx_n \notin A$ leads to a contradiction. We analyze two cases.

Let $n = 0$.

$$
\begin{aligned}
& \Diamond r^* t x_n \in A && \\
\Longrightarrow\ & \exists \alpha \in \mathcal{V}A\colon \alpha = \Diamond r^* t, \alpha x_n \in A && \mathcal{M}_{\Diamond^*} \\
\Longrightarrow\ & \exists \alpha \in \mathcal{V}A\colon \Diamond r \alpha x_n \in A && \mathcal{M}_\alpha, tx \notin A \\
\Longrightarrow\ & \exists y \in \mathcal{N}_\iota A\colon x_n \xrightarrow{\trianglerighteq, \Diamond r^* t}_A y && \mathcal{M}_\Diamond
\end{aligned}
$$

Let $n > 0$.

$$
\begin{array}{lll}
& x_{n-1} \xrightarrow{\trianglerighteq, \Diamond r^* t}_A x_n & \text{Def. 50} \\
\iff & \exists x' \in \mathcal{N}_\iota A \colon x_{n-1} \trianglerighteq_A^{\Diamond r^* t} x' \wedge x' \xrightarrow{\Diamond r^* t}_A x_n & \text{Def. 48} \\
\iff & \exists x' \in \mathcal{N}_\iota A \colon x_{n-1} \trianglerighteq_A^{\Diamond r^* t} x' & \\
& \wedge \exists \alpha \in \mathcal{V}A \colon \alpha = \Diamond r^* t, \alpha x', r x' x_n, \alpha x_n \in A & \text{Def. 21, Def. 20} \\
\implies & \exists \alpha \in \mathcal{V}A \colon \alpha = \Diamond r^* t, \Diamond r \alpha x_n \in A & \mathcal{M}_\alpha, t x_n \notin A \\
\implies & \exists y \in \mathcal{N}_\iota A \colon x_n \xrightarrow{\trianglerighteq, \Diamond r^* t}_A y & \mathcal{M}_\Diamond
\end{array}
$$

In both cases we have a contradiction, since $(x_0, \ldots, x_n)$ is $\xrightarrow{\trianglerighteq, \Diamond r^* t}_A$-maximal by Def. 50.□

Having established this lemma, it is easy to formalize a condition to identify branches which may lack some witnesses. All we need is a cycle in the $\Diamond^*$-request relation.

Definition 51 (Loop) Let A be an admissible branch. A **loop** is a request path $(x_0, \ldots, x_n)$ such that $x_0 = x_n$ and $n > 0$.

Proposition 39 Let A be a maximal, admissible branch and $x \in \mathcal{N}_\iota A$. If A is loop-free, then A is evident.

Proof. Since A is maximal, it satisfies all evidence conditions except (possibly) $\mathcal{E}_{\Diamond^*}$. Let $\Diamond r^* t x \in A$. Since A is loop-free, no request path for in A is cyclic. By Lem. 6, $\Diamond r^* t x$ is evident in A. □

The last proposition allows us to identify evident branches by checking that no loops occur.

6.3.4 Detecting Loops

Checking for loops requires some computation. In the following we give a criterion which is easier to check.

Proposition 40 Let A be a branch and $\alpha = \Diamond r^* t \in A$. Then $\xrightarrow{\alpha}_A \subseteq \xrightarrow{\trianglerighteq, \Diamond r^* t}_A$.

The idea is to track which formulas $\Diamond r \alpha x$ were blocked by an expanded formula $\Diamond r \beta y$. In this case we say **α is blocked by β**. If β then is blocked by α again, we have a cycle in the corresponding $\Diamond^*$-request path.

First we refine the pattern equivalence relation to only relate $\xrightarrow{\alpha}_A$-terminal nominals x to other nominals if $\Diamond r \alpha x$ is on the branch, i.e. x demands a successor satisfying α.

Definition 52 (Blocking Nominal Relation)

$$
x \triangleright_A^{\alpha, r} x' \iff x \sim_A^r x' \wedge \Diamond r \alpha x \in A \wedge \neg \exists y \in \mathcal{N}_\iota A \colon x \xrightarrow{\alpha}_A y
$$

We are now ready to formulate the blocking relation among path variables. We relate a path variable α to a path variable β, if α was blocked by β.

Definition 53 (Blocking Path Variable Relation)

$$\alpha \rhd_A \beta \iff \exists r, t\colon \alpha = \Diamond r^* t, \beta = \Diamond r^* t \in A$$
$$\land\, \exists x, x', y \in \mathcal{N}_\iota A\colon x \rhd_A^{\alpha,r} x' \land x' \xrightarrow{\beta}_A y$$

We now prove that a cycle in $\rhd_A$ corresponds to a cycle in the corresponding $\Diamond^*$-request relation.

Lemma 7 Let A be an admissible branch, $\alpha = \Diamond r^* t \in A$, $\beta \in \mathcal{V}A$, and $\alpha \rhd_A \beta$. Then $\exists y \in \mathcal{N}_\iota A\colon \beta y \in A \land \forall x \in \mathcal{N}_\iota A\colon \alpha x \in A \implies x \xrightarrow{\unrhd, \Diamond r^* t^*}_A y$.

Proof.

$$\begin{aligned}
& \alpha \rhd_A \beta \\
\iff & \exists r, t\colon \alpha = \Diamond r^* t, \beta = \Diamond r^* t \in A \\
& \land\, \exists x, x', y \in \mathcal{N}_\iota A\colon x \rhd_A^{\alpha,r} x' \land x' \xrightarrow{\beta}_A y \\
\iff & \exists r, t\colon \alpha = \Diamond r^* t, \beta = \Diamond r^* t \in A \land \exists x, x', y \in \mathcal{N}_\iota A\colon x \sim_A^r x' \\
& \land\, \Diamond r \alpha x \in A \land \neg \exists y' \in \mathcal{N}_\iota A\colon x \xrightarrow{\alpha}_A y' \land x' \xrightarrow{\beta}_A y
\end{aligned}$$

We get $\alpha x \in A$ by admissibility of A from $\Diamond r \alpha x \in A$. Since the set $\{x \mid \alpha x \in A\}$ is an α-path by admissibility of A, and it holds $\neg \exists y \in \mathcal{N}_\iota A\colon x \xrightarrow{\alpha}_A y$, we have that x is the last element of the path. Thus by Prop. 40, we have $\forall x' \in \mathcal{N}_\iota A\colon \alpha x' \in A \implies x' \xrightarrow{\unrhd, \Diamond r^* t^*}_A x$. The claim follows. □

Proposition 41 Let A be an admissible branch. If $\rhd_A$ is cyclic, there is a $\xrightarrow{\unrhd, \Diamond r^* t}_A$-loop.

Proof Sketch. By Induction on the length of the cycle in $\rhd_A$. □

6.4 Completeness of $\mathcal{T}_{pat}$

We now prove completeness of $\mathcal{T}_{pat}$. This means we have to show that for every satisfiable initial branch we end up with a maximal tableau that contains at least one evident branch. This situation is complicated by the fact that we have three kinds of maximal branches, instead of the usual two. From straightness of $\mathcal{T}$ it follows that $\mathcal{T}_{pat}$ is also straight. We will use this result to show that if the initial branch A is satisfiable, there is one evident, straight branch in every maximal tableau for A.

We have already shown that the function that assigns each nominal to its pattern is a selector in Prop. 9. We now seek to lift Prop. 15 to the relation $\xrightarrow{\unrhd, \Diamond r^* t}_A$. This will enable us to prove that $\xrightarrow{\unrhd, \Diamond r^* t}_A$ is also loop-free.

Lemma 8 Let A be an admissible, straight branch, and $\mathcal{I}$ be straight for A. If $x \xrightarrow{\trianglerighteq,\Diamond r^*t}_A y$ and $\mathcal{I} \not\models ty$, then $\Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A x) > \Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A y)$.

Proof.

$$
\begin{array}{lll}
& x \xrightarrow{\trianglerighteq,\Diamond r^*t}_A y & \\
\iff & \exists x' \in \mathcal{N}_\iota A \colon x \trianglerighteq^{\Diamond r^*t}_A x' \wedge x' \xrightarrow{\Diamond r^*t}_A y & \text{Def. 48} \\
\implies & \exists x' \in \mathcal{N}_\iota A \colon x \sim^r_A x' \wedge x' \xrightarrow{\Diamond r^*t}_A y & \text{Def. 47} \\
\implies & \exists x' \in \mathcal{N}_\iota A \colon x \sim^r_A x' \wedge \Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A x') > \Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A y) & \text{Prop. 15, Prop. 9} \\
\implies & \exists x' \in \mathcal{N}_\iota A \colon \mathcal{P}^r_A x = \mathcal{P}^r_A x' \wedge \Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A x') > \Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A y) & \text{Def. 16} \\
\implies & \Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A x) > \Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A y) & \square
\end{array}
$$

We have now established that the witness distance decreases also along $\xrightarrow{\trianglerighteq,\Diamond r^*t}_A$ on straight branches, provided that the witness was not possible at the origin. We now prove that the witness is never possible at the origin. The key is that in Def. 47 we demand a request to be present at all non-terminal nodes on the witness path.

Lemma 9 Let A be an admissible, straight branch, $\mathcal{I}$ be a modal interpretation that is straight for A, and $x, y \in \mathcal{N}_\iota A$. If $x \xrightarrow{\trianglerighteq,\Diamond r^*t}_A y$, then $\mathcal{I} \not\models tx$.

Proof. By Def. 48 and Def. 47, there is $\alpha \in \mathcal{V}A$ such that $\alpha = \Diamond r^*t, \Diamond r\alpha x \in A$. By O2, $\mathcal{I} \not\models tx$. □

Now comes the crucial lemma of this section. We prove that no pattern can repeat before the witness is created.

Lemma 10 Let A be a propagated branch. If A is straight, then A is loop-free.

Proof. Since A is straight, there is a modal interpretation $\mathcal{I}$ that is straight for A. Assume $(x_0, \dots, x_n)$ is a loop in A for contradiction. By Lem. 9, $\mathcal{I} \not\models tx_i$ for all $i \in [0, n-1]$. By Def. 51 it holds $x_0 = x_n$. Thus, $\mathcal{I} \not\models tx_n$. Lem. 8 yields $\Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A x_0) > \Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A x_n)$. But $\Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A x_0) = \Delta^r_{\mathcal{I},t}(\mathcal{P}^r_A x_n)$, since $\Delta^r_{\mathcal{I},t}$ and $\mathcal{P}^r_A$ are functions. Contradiction. □

Theorem 4 (Completeness) Every maximal $\mathcal{T}_{pat}$-tableau for a satisfiable set of K*-expressions contains an evident branch.

Proof. For every satisfiable set of K*-expressions L there is a straight initial branch, namely $\{tx_0 \mid t \in L\}$. Since L by construction contains no edges and no path variables, any modal interpretation satisfying L is straight for L. A maximal tableau for L contains a maximal branch A which is straight by Prop. 16. By Lem. 10, A is loop-free. By Prop. 39, A is evident. □

7 Conclusion

The logic K^* includes an operator for the reflexive transitive closure, and is not compact. Maximal tableaux for K^* may contain branches that are neither evident nor closed.

To solve the problem of discarding such branches, straightness is established for the tableau system. Straightness is similar to soundness, in that it is sustained by the tableau rules for at least one branch among the alternatives. We prove that on satisfiable, straight branches witnesses occur before requests repeat.

We give two terminating tableau systems for K^*, and prove both complete. The key is to discard branches with cyclic request relations. Verification soundness is established by showing that branches with acyclic request relations are evident. Refutation soundness is established by showing that on straight branches request relations are never cyclic. This idea works for both blocking techniques under consideration.

Improvements

The proof of straightness in Chapter 4 used an interpretation that satisfies all formulas except the edges on the branch. We could not require the interpretation to satisfy the edges, since we cannot guarantee that we have a direct successor for every $\Diamond$-formula that provides the minimal witness distance in the interpretation. We solve the problem using a selection technique from Baader to find a suitable state among all states in the interpretation, using additional conditions to ensure satisfaction of $\Box$-formulas.

To make the proof more transparent, it is desirable to have the interpretation satisfy the edges as well. Using a canonical interpretation in the proof of straightness, our conjecture is, that a canonical interpretation can also satisfy the edges. We must ensure that there are sufficient direct successors to provide every eventuality with the optimal witness distance. This requires the set of boxes present in the interpretation to be exactly the set of boxes of the associated nominal present on the branch. Our conjecture is, that this is possible using a canonical interpretation and a carefully selected initial state. The straightness invariants can then be maintained under $\Diamond$-expansion by carefully selecting the state among the successors in the interpretation. Our conjecture is that in canonical interpretations it is sufficient to choose the state among the successors that have the minimal witness distance such that the size of the request of the state is minimal.

The abstract concept behind our completeness proof is interesting. Terminating tableau systems only produce a subset of the models of a logic. The key insight is that we must identify a subset of the models produced by the tableau that is complete for the logic in the sense that for every satisfiable formula of the logic there is a model in the class. Completeness for the system can then be proven on the basis of properties of the complete class. We conjecture that this process works the other way round as well. Given a complete class of models, it should be possible to devise a blocking technique tailored for this class. If the conjecture is true, it should be the case to determine the most restrictive blocking technique with respect to a class of models.

One step towards a minimal model class for K^* are the canonical interpretations. At the end of Chapter 4 we have been able to show that canonical models satisfy a condition analogous to straightness: Given an expression $\Diamond r(\Diamond r^* t)$ in the saturation closure of the expression a canonical interpretation is built for, every state in the canonical interpretation that satisfies $\Diamond r(\Diamond r^* t)$ also has an immediate r-successor with minimal witness distance for t.

Outlook

The next step is to extend this proof to full PDL. One would need to take the Fischer-Ladner closure instead of the extended subterm closure, and adapt the termination proof accordingly. The other proofs also have to be reworked, but it seems reasonable to assume that the approach scales to full PDL.

The investigation of canonical interpretations for the proof of straightness, as described before, is also a possible topic of interest. The main difficulty is to maintain the straightness invariants under $\Diamond$-expansion, which requires a selection technique.

Finally, we consider it promising to investigate wether the pattern-based blocking conditions can be reformulated on the basis of requests. This requires two main steps. First, a model existence theorem based on a suitable reachability relation must be proven. And second, a suitable realization condition based on requests must be formulated, and the completeness proof must be adapted.

Bibliography

[1] P. Abate, R. Goré, and F. Widmann. An on-the-fly tableau-based decision procedure for PDL-satisfiability. In C. Areces and S. Demri, editors, *Proceedings of the 5th Workshop on Methods for Modalities (M4M5 2007)*, volume 231 of *Electronic Notes in Theoretical Computer Science*, pages 191–209. Elsevier, 2009.

[2] P. B. Andrews. *An Introduction to Mathematical Logic and Type Theory: To Truth Through Proof*, volume 27 of *Applied Logic Series*. Kluwer Academic Publishers, second edition, 2002.

[3] F. Baader. Augmenting concept languages by transitive closure of roles: An alternative to terminological cycles. In J. Mylopoulos and R. Reiter, editors, *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI 1991)*, volume 1, pages 446–451. Morgan Kaufmann, 1991.

[4] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2001.

[5] T. Bolander and P. Blackburn. Termination for hybrid tableaus. *Journal of Logic and Computation*, 17(3):517–554, 2007.

[6] A. Church. A formulation of the simple theory of types. *Journal of Symbolic Logic*, 5(1):56–68, 1940.

[7] W. M. Farmer. The seven virtues of simple type theory. *Journal of Applied Logic*, 6(3):267–286, 2008.

[8] M. J. Fischer and R. E. Ladner. Propositional modal logic of programs (extended abstract). In *Conference Record of the Ninth Annual ACM Symposium on Theory of Computing (STOC 1977)*, pages 286–294. ACM, 1977.

[9] G. D. Giacomo and F. Massacci. Combining deduction and model checking into tableaux and algorithms for converse-pdl. *Information and Computation*, 162(1-2):117–137, 2000.

[10] R. Goldblatt. Mathematical modal logic: A view of its evolution. *Journal of Applied Logic*, 1(5-6):309–392, 2003.

[11] R. Goré and F. Widmann. An optimal on-the-fly tableau-based decision procedure for pdl-satisfiability. In R. A. Schmidt, editor, *Proceedings of the 22nd International Conference on Automated Deduction (CADE 2009)*, volume 5663 of *Lecture Notes in Computer Science*, pages 437–452. Springer, 2009.

[12] D. Götzmann. Spartacus: A tableau prover for hybrid logic. M.Sc. Thesis, Programming Systems Lab, Department of Computer Science, Saarland University, January 2009.

[13] D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. MIT Press, 2000.

[14] C. A. R. Hoare. An axiomatic basis for computer programming. *Communications of the ACM*, 12(10):576–580, 1969.

[15] I. Horrocks and U. Sattler. A tableau decision procedure for $\mathcal{SHOIQ}$. *J. Autom. Reasoning*, 39(3):249–276, 2007.

[16] G. E. Hughes and M. J. Cresswell. *A New Introduction to Modal Logic*. Routledge, September 1996.

[17] M. Kaminski, S. Schneider, and G. Smolka. Terminating tableaux for graded hybrid logic with global modalities and role hierarchies. In M. Giese and A. Waaler, editors, *Proceedings of the 18th International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX 2009)*, volume 5607 of *Lecture Notes in Computer Science (LNAI)*, pages 235–249. Springer, 2009.

[18] M. Kaminski and G. Smolka. Terminating tableaux for hybrid logic with the difference modality and converse. In A. Armando, P. Baumgartner, and G. Dowek, editors, *Automated Reasoning, 4th International Joint Conference (IJCAR 2008)*, volume 5195 of *Lecture Notes in Computer Science (LNAI)*, pages 210–225. Springer, 2008.

[19] M. Kaminski and G. Smolka. Hybrid tableaux for the difference modality. In C. Areces and S. Demri, editors, *Proceedings of the 5th Workshop on Methods for Modalities (M4M5 2007)*, volume 231 of *Electronic Notes in Theoretical Computer Science*, pages 241–257. Elsevier, 2009.

[20] M. Kaminski and G. Smolka. Terminating tableau systems for hybrid logic with difference and converse. *Journal of Logic, Language and Information*, 2009. To appear.

[21] M. Kaminski and G. Smolka. Terminating tableaux for SOQ with number restrictions on transitive roles. In B. C. Grau, I. Horrocks, B. Motik, and U. Sattler, editors, *Proceedings of the 22nd International Workshop on Description Logics (DL 2009)*, volume 477 of *CEUR Workshop Proceedings*. CEUR-WS.org, July 2009.

[22] S. A. Kripke. A completeness theorem in modal logic. *Journal of Symbolic Logic*, 24(1):1–14, 1959.

[23] S. A. Kripke. Semantic analysis of modal logic I: Normal modal propositional calculi. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, 9:67–96, 1963.

[24] S. A. Kripke. Semantical considerations on modal logic. *Acta Philosophica Fennica*, 16:83–94, 1963.

[25] C. I. Lewis. *A Survey of Symbolic Logic*. University of California Press, 1918.

[26] V. R. Pratt. Semantical considerations on floyd-hoare logic. In *Proceedings of the 17th Annual Symposium on Foundations of Computer Science (FOCS 1976)*, pages 109–121. IEEE, 1976.

[27] G. Smolka. *Introduction to Computational Logic: Lecture Notes*. Universität des Saarlandes, 2009. www.ps.uni-sb.de/courses/cl-ss09/script/.