Term-Modal Logics

Melvin Fitting[1], Lars Thalmann[2], and Andrei Voronkov[3]

[1] City University of New York
fitting@alpha.lehman.cuny.edu
[2] Uppsala University
talman@csd.uu.se
[3] University of Manchester
voronkov@cs.man.ac.uk

Abstract. Many powerful logics exist today for reasoning about multi-agent systems, but in most of these it is hard to reason about an infinite or indeterminate number of agents. Also the naming schemes used in the logics often lack expressiveness to name agents in an intuitive way.
To obtain a more expressive language for multi-agent reasoning and a better naming scheme for agents, we introduce a family of logics called *term-modal logics*. A main feature of our logics is the use of modal operators indexed by the terms of the logics. Thus, one can *quantify over variables occurring in modal operators.* In term-modal logics agents can be represented by terms, and knowledge of agents is expressed with formulas within the scope of modal operators.
This gives us a flexible and uniform language for reasoning about the agents themselves and their knowledge. This paper gives examples of the expressiveness of the languages and provides sequent-style and tableau-based proof systems for the logics. Furthermore we give proofs of soundness and completeness with respect to the possible world semantics.

1 Introduction

1.1 Term-modal logics

In this paper, we describe a new family of modal logics, namely the term-modal first-order logics, where we by *term-modal* mean that any term can be used as a modality. The specific logics we discuss are the term-modal versions of the modal logics $\mathsf{K}, \mathsf{K4}, \mathsf{D}, \mathsf{D4}, \mathsf{T}$ and $\mathsf{S4}$. Sequent-style and tableau-style proof systems for the logics are given, and their soundness and completeness are shown.

1.2 Motivations

Many researchers have been interested in the use of multi-modal logics for knowledge representation, although most of them have been investigating the use of a finite set of modalities, indexed by the first n natural numbers, usually denoted either $[1], [2], \ldots, [n]$ or $\mathcal{K}_1, \mathcal{K}_2, \ldots, \mathcal{K}_n$. When instead using an infinite set of modalities we can reason about a dynamic society of agents, where some agents might vanish and new agents may join the group.

In the family of multi-modal logics presented in this paper, any term can denote an agent. This makes naming of agents easy and the logics expressive. The use of complex names, possibly involving variables, for agents makes it easy to model a society of agents, and give names to new agents by their relationship to already existing agents. For example, to express that the agent $mother(x)$ thinks (or knows, or believes) that the agent x is good, we can write $[mother(x)]good(x)$.

The standard multi-modal logics allow us to reason about beliefs of particular agents, but provide very limited facilities to reason about believes of groups of agents or agents themselves. In our language, we can distinguish a group of agents by specifying their properties. For example, to express that every Christian believes in the existence of God, we can write $\forall x(christian(x) \supset [x]\exists y\, God(y))$.

An example of a society of agents is the collection of computer processes running on some system. Here the logic with its infinite complex naming mechanism can be used to specify requirements of the system as a whole and the proof system can be used to check that these requirements are satisfied.

When the computer processes spawn other processes, the society of agents (i.e. the number of processes) grows, and the naming mechanism can be used to refer to the newly created processes. As the number of processes spawned by the program may not be known beforehand, it is convenient to have an unlimited set of names for these new agents.

Many researchers have investigated multi-modal logic with a finite set of modalities, and many have discussed naming. We here combine these two aspects into one general family of logics, the family of term-modal logics.

1.3 Background

In Fitting [3], proofs of soundness and completeness of (single-)modal logics are given. In this paper we introduce some new definitions, and extend the proofs for the term-modal logics.

Fagin et al. [2] use modal logics to describe multi-agent systems. Their approach is based on a finite set of agents, and they also discuss the use of common and distributed knowledge. By using the logic presented in this paper, their work might be extended to handle dynamic agent societies with an easy naming mechanism, where quantification over agents is possible.

1.4 Overview

The rest of this paper is structured as follows. Section 2 defines the syntax of term-modal logics, and Section 3 their semantics. In Section 4 we introduce sequent calculi and in Section 5 tableau calculi for these logics. Soundness and completess of the sequent calculi are proved in Fitting, Thalmann, and Voronkov [5]. As a step toward automated reasoning in term-modal logic in Section 6 we introduce free-variable versions of tableau calculi for term-modal logics. In Section 7 we give an example refutation in such a calculus, in order to illustrate some distinctive features of free-variable calculi for term-modal logics.

Proofs not included in this paper can be found in Fitting, Thalmann, and Voronkov [5].

2 Syntax

The term-modal logics are obtained from the standard predicate modal logics by adding modal operators indexed by terms. In this section we give a formal definition of the syntax of term-modal logics.

We assume a *signature* Σ consisting of three disjoint sets of constants, function symbols and relation symbols. Usually, the signature is assumed to be fixed, but in some situation we will vary it for technical convenience. In addition to the symbols of Σ, we will use various infinite *sets* P *of parameters* disjoint from the symbols in Σ. In some situations parameters will behave as new constants, in others as elements of a domain on which formulas are evaluated. The signature is not necessarily finite or countable. For every signature Σ, we denote by Σ^- the signature obtained from Σ by omitting all constants and function symbols.

Definition 1 (term) Suppose P is a set of parameters and V a set of *variables* disjoint from the set of parameters. The set of *terms of the signature* Σ *with parameters in* P and variables in V, denoted $\mathcal{T}(\Sigma \cup P, V)$ is defined inductively as follows.

1. Each constant in Σ, variable in V, and symbol in P is a term.
2. If $t_1, \ldots, t_n$ are terms and f is an n-ary function symbol, then $f(t_1, \ldots, t_n)$ is a term.

In this paper, we can restrict ourselves to a fixed set of variables, however the set of parameters (and sometimes the signature) will vary. So we will use a simpler notation $\mathcal{T}(\Sigma \cup P)$. A term is called *ground* if it has no occurrences of variables.

Definition 2 (formula) Let P be a set of parameters. The set of *formulas of the signature* Σ *with parameters in* P, denoted $\mathcal{F}(\Sigma \cup P)$, is defined inductively as follows.

1. If R is a relation symbol of arity n and $t_1, \ldots, t_n$ are terms in $\mathcal{T}(\Sigma \cup P)$, then $R(t_1, \ldots, t_n)$ is an *atomic formula*.
2. If A and B are formulas, then so are $(A \wedge B)$, $(A \vee B)$, $(A \supset B)$ and $\neg A$.
3. If A is a formula and t is a term in $\mathcal{T}(\Sigma \cup P)$, then $[t]A$ and $\langle t \rangle A$ are formulas.
4. If A is a formula and x is a variable, then $\forall x A$ and $\exists x A$ are formulas.

The notions of *free and bound occurrences of variables* are defined as usual, with the exception of the following item:

- The free occurrences of variables in $[t]A$ and $\langle t \rangle A$ are all occurrences of variables in t plus all free occurrences of variables in A.

A formula is called *closed*, or a *sentence* if it has no free occurrences of variables (but note that it can contain parameters). A $\exists$-*formula* is any formula $\exists x A$. A *literal* is either an atomic formula A or its negation $\neg A$. Literals A and $\neg A$ are called *complementary* to each other.

Intuitively, when interpreting the formulas in a multi-agent context, the meaning of the formula $[t]A$ is that the agent denoted by the term t knows (believes etc.) the information represented by the formula A. The formula $\langle t\rangle A$ intuitively means that the agent denoted by t considers it possible that A holds, i.e., it is not the case that the agent knows the contrary (which can also be expressed by $\neg[t]\neg A$).

In the proof systems introduced later we make no assumptions about the kind of knowledge expressed. The knowledge could in fact be just beliefs, i.e., an agent might believe something which is false.

It is easy to add axioms of knowledge, if one is interested in describing a specific kind of knowledge. An example of this is the knowledge axiom, $[x]A \supset A$, which intuitively means that if an agent x knows something, then it is true. More about different epistemic interpretations of modal logic can be found in e.g. Hintikka [11].

The distinctive feature of our logics is the possibility to express knowledge of agents and properties of agents themselves in one language. For example, we can write $\forall x(\mathit{human}(x) \supset [x]\mathit{good}(x))$ to express that everyone knows that he/she is good. Note the use of quantification over agents.

For the rest of this paper, we will denote

- variables by x, y, z, u, v;
- terms by s, t;
- domain elements by d (i.e. elements in the set $\mathbf{D}$, which is defined later);
- formulas by A, B, C;
- sets of parameters of the language by P;
- parameters by p;
- literals by L;
- logics $\mathsf{K}, \mathsf{K4}, \mathsf{D}, \mathsf{D4}, \mathsf{T}, \mathsf{S4}$ by the generic symbol $\mathcal{L}$.

We write $A(x)$ to denote a formula A with (zero or more) free occurrences of the variable x and write $A(t)$ to denote the replacement of all free occurrences of x by a term t. Before the replacement, we rename in $A(x)$ all bound occurrences of variables that have free occurrences in t.

2.1 Related work

The current trend in modal and description logics is to define expressive but still decidable logics. Our logics are undecidable since they contain first-order classical logic. Moreover, the expressiveness of our logics is, in a way, higher than that of the standard first-order modal logics, since first-order modal logics can be interpreted in our logics by using a single constant in modal operators (at least for cumulative domains, but our results can be extended to the constant domain versions of the logics as well). Logics with modalities indexed by terms were studied by Grove and Halpern [10, 9]. These logics are more expressive in some aspects and less expressive in other aspects than ours. Namely, these logics can handle equality and agents with special properties. However, there are restrictions on how formulas can be built in these logics, so some well-formed formulas of our logics cannot be used as formulas in Grove and Halpern [10, 9].

3 Semantics

3.1 Frames

In this section, we describe a possible world semantics for the logics. The semantics is defined through the notions of *frames* and *structures*. It differs from the standard semantics of first-order modal logics by the treatment of the reachability relation on worlds: the reachability relation is indexed by elements of the domain. We assume all definitions be given w.r.t. a nonempty set $\mathbf{D}$, called the *domain*.

DEFINITION 3 (frame) A *frame* over $\mathbf{D}$ is a triple $\langle \mathcal{W}, \mathcal{D}, \longrightarrow \rangle$, where

1. $\mathcal{W}$ is a non-empty set, called the *set of possible worlds*.
2. $\mathcal{D}$ is a mapping from $\mathcal{W}$ to the set of subsets of $\mathbf{D}$. The set $\mathcal{D}(w)$ is denoted by $\mathcal{D}_w$ and called the *domain of* w.
3. $\longrightarrow$ is a relation on $\mathcal{W} \times \mathbf{D} \times \mathcal{W}$, called the *accessibility relation*. If $\longrightarrow (w_1, d, w_2)$, then we say that w_2 is d*-reachable* from w_1 and write $w_1 \xrightarrow{d} w_2$.

We require the following *monotonicity condition* to be satisfied in all frames:[1] if $w_1 \xrightarrow{d} w_2$, then $\mathcal{D}_{w_1} \subseteq \mathcal{D}_{w_2}$. The monotonicity condition corresponds to *cumulative domains*, see e.g. Wallen [15] (also known as *nested domains*, see Garson [6]).

DEFINITION 4 ($\mathcal{L}$-frame) We specialize the concept of frames to six different classes:

- K. All frames are K*-frames*.
- T. If $w \xrightarrow{d} w$ holds for all w and d (i.e. the accessibility relation is reflexive in its 1st and 3rd arguments), then the frame is a T*-frame*.
- D. If for all d and w there exists w' such that $w \xrightarrow{d} w'$ (i.e. the accessibility relation satisfies seriality in its 1st and 3rd arguments) then the frame is a D*-frame*.
- K4. If from $w \xrightarrow{d} w'$ and $w' \xrightarrow{d} w''$ it follows that $w \xrightarrow{d} w''$ for all d and $w, w', w'' \in \mathcal{W}$, (i.e. the accessibility relation is transitive in its 1st and 3rd arguments) then it is a K4*-frame*.
- S4. If the accessibility relation is both reflexive and transitive in its 1st and 3rd arguments, then the frame is an S4*-frame*.
- D4. If the accessibility relation is both reflexive and satisfies seriality in its 1st and 3rd arguments, then the frame is a D4*-frame*.

[1] The monotonicity condition can be replaced by a weaker condition: if $w_1 \xrightarrow{d} w_2$ and $d \in w_1$ then $\mathcal{D}_{w_1} \subseteq \mathcal{D}_{w_2}$. The reason is that the first-order language cannot express properties of worlds d-reachable from w, when $d \notin \mathcal{D}_w$.

3.2 First-order modal structures

The first-order modal (Kripke) structures are introduced in the standard way, except for the case of modal operators.

DEFINITION 5 (structure for $\mathcal{L}$) Let $\mathcal{L}$ be one of K, T, D, K4, S4, D4. A *first-order modal structure* for $\mathcal{L}$, or simply *$\mathcal{L}$-structure* over a domain $\mathbf{D}$ is a tuple $\mathfrak{S} = \langle \mathcal{W}, \mathcal{D}, \longrightarrow, I, \Vdash \rangle$, where

1. $\langle \mathcal{W}, \mathcal{D}, \longrightarrow \rangle$ is an $\mathcal{L}$-frame over $\mathbf{D}$.
2. $\Vdash$ is a binary relation between worlds and atomic sentences in $\mathcal{F}(\Sigma^- \cup \mathbf{D})$. (Note that elements of $\mathbf{D}$ are treated as parameters in $\mathcal{F}(\Sigma^- \cup \mathbf{D})$).
3. I, called the *interpretation function*, is a mapping that maps every constant c of Σ to an element of $\mathbf{D}$ and every function symbol f of Σ of arity n to an n-place function on $\mathbf{D}$. The corresponding element of $\mathbf{D}$ and function on $\mathbf{D}$ are called the *interpretations* of c and f respectively. We require the interpretation of any constant and function symbol to be totally defined in every world: this means that $I(c)$ belongs to $\mathcal{D}_w$ for every world $w \in \mathcal{W}$ and for every $d_1, \ldots, d_n \in \mathcal{D}_w$ we have $I(f)(d_1, \ldots, d_n) \in \mathcal{D}_w$.

Note that $\Vdash$ is only defined on formulas without function symbols or constants, but with parameters in $\mathbf{D}$.

We call a *valuation V in a structure* $\mathfrak{S}$ any mapping $V : P \to \mathbf{D}$ from a set of parameters to the domain $\mathbf{D}$ of $\mathfrak{S}$. Any valuation V can be extended to the set of all ground terms by defining

$$\begin{aligned} V(c) &= I(c); \\ V(f(t_1, \ldots, t_n)) &= I(f)(V(t_1), \ldots, V(t_n)). \end{aligned}$$

Now we can give the central notion of satisfiability of formulas in structures. Given a first-order modal structure $\langle \mathcal{W}, \mathcal{D}, \longrightarrow, I, \Vdash \rangle$, we change the relation $\Vdash$ into a ternary relation between worlds in $\mathcal{W}$, valuations, and sentences in $\mathcal{F}(\Sigma \cup \mathbf{D})$ as given below. We write $\mathfrak{S}, w, V \Vdash A$ when this relation holds on $\mathfrak{S}, w, V, A$ and denote by $\nVdash$ the complement of $\Vdash$. When we use this notation, we can omit one or both of $\mathfrak{S}, V$, when they are clear from the context.

DEFINITION 6 (relation $\Vdash$) Given $\mathfrak{S}$ and V, we define the relation $\Vdash$ as follows.

1. $w, V \Vdash R(t_1, \ldots, t_n)$ if $w \Vdash R(V(t_1), \ldots, V(t_n))$.
2. $w, V \Vdash A \wedge B$ if $w, V \Vdash A$ and $w, V \Vdash B$.
3. $w, V \Vdash A \vee B$ if $w, V \Vdash A$ or $w, V \Vdash B$.
4. $w, V \Vdash A \supset B$ if $w, V \nVdash A$ or $w, V \Vdash B$.
5. $w, V \Vdash \neg A$ if $w, V \nVdash A$.
6. $w, V \Vdash [t]A$ if for all w' such that $w \xrightarrow{V(t)} w'$ we have $w', V \Vdash A$.
7. $w, V \Vdash \langle t \rangle A$ if there exists w' such that $w \xrightarrow{V(t)} w'$ and $w', V \Vdash A$.
8. $w, V \Vdash \forall x A(x)$ if $w, V \Vdash A(d)$, for all $d \in \mathcal{D}_w$.
9. $w, V \Vdash \exists x A(x)$ if $w, V \Vdash A(d)$, for some $d \in \mathcal{D}_w$.

Definition 7 (truth, satisfiability)
Let $\mathfrak{S} = \langle \mathcal{W}, \mathcal{D}, \longrightarrow, I, \Vdash \rangle$ be a structure. We say a sentence A is *true*, or *holds*, or *is locally satisfied* in $\mathfrak{S}$ at a world $w \in \mathcal{W}$ *under a valuation* V if $\mathfrak{S}, w, V \Vdash A$. A formula A is *globally satisfied* in a structure $\mathfrak{S}$ under a valuation V if it is satisfied at every world of $\mathfrak{S}$ under V. A formula A is called *locally (respectively, globally) satisfiable in* $\mathfrak{S}$ if it is locally (respectively, globally) satisfied in $\mathfrak{S}$ under some valuation. If A is locally satisfiable in $\mathfrak{S}$ we also say that $\mathfrak{S}$ is a *model* of A.

Note that the truth of a formula A under a valuation V only depends on the value of V on the parameters occurring in A. Thus, if A is a sentence of Σ, its truth does not depend on the valuation at all.

Definition 8 ($\mathcal{L}$-model, validity) Let $\mathcal{L}$ be one of K, T, D, $\mathsf{K4}$, $\mathsf{S4}$, $\mathsf{D4}$. We call a model $\langle \mathcal{W}, \mathcal{D}, \longrightarrow, I, \Vdash \rangle$ of a formula A an *$\mathcal{L}$-model* if its frame $\langle \mathcal{W}, \mathcal{D}, \longrightarrow \rangle$ is an $\mathcal{L}$-frame. A formula A is called *$\mathcal{L}$-satisfiable* if it has an $\mathcal{L}$-model. A formula A is called *$\mathcal{L}$-valid* if it is true in every world of every $\mathcal{L}$-structure under every valuation.

It is not hard to argue that satisfiability and validity are dual notions in the following sense: a formula A is unsatisfiable if and only if $\neg A$ is valid. In view of this duality we will formulate our results in terms of (un)satisfiability only.

When we speak of a *logic* $\mathcal{L}$ in this paper, we understand the set of $\mathcal{L}$-valid formulas. So, we will speak of *logics* K, T, D, $\mathsf{K4}$, $\mathsf{S4}$, $\mathsf{D4}$. Another standard way of introducing a logic is to define a suitable *calculus* deriving valid formulas in this logic. In the next section we introduce such calculi for all these logics.

Formulas A and B are called *$\mathcal{L}$-equivalent* if the formulas $A \supset B$ and $B \supset A$ are valid. It is evident that in any context when we speak about worlds, structures, valuations, and satisfiability, we can replace formulas by equivalent ones.

We will now introduce the negation normal form of formulas, which will simplify our proofs considerably.

Definition 9 (negation normal form) A formula A is said to be in *negation normal form* if it is constructed from literals using $\wedge$, $\vee$, $\forall$, $\exists$, $[t]$ and $\langle t \rangle$. A formula B is called a *negation normal form of a formula* A, if B is in negation normal form and B is equivalent to A.

It is not hard to argue that every formula has a negation normal form, so we will only consider formulas in negation normal form.

4 Sequent calculi

In this section we define sequent calculi for the family of term-modal logics. There are several essentially equivalent notions of sequent giving rise to different calculi. The original definition of Gentzen [7] defines sequents as expressions $A_1, \ldots, A_n \rightarrow B_1, \ldots, B_m$, where $A_1, \ldots, A_n, B_1, \ldots, B_m$ are formulas.

Smullyan [13] represents such a sequent as a collection $A_1, \dots, A_n, \neg B_1, \dots, \neg B_m$ of formulas or a collection $T\ A_1, \dots, T\ A_n, F\ B_1, \dots, F\ B_m$ of *signed formulas* with the intended meaning that all formulas A_i are true and all formulas B_j are false and introduces a *uniform notation* to group together inference rules with similar behavior. We will use the approach due to Schütte [12]. Instead of using arbitrary formulas we will use only formulas in *negation normal form*. Every rule in the uniform notation corresponds to an inference rule introducing a particular connective on formulas in negation normal form. Then we do not need the unifying notation anymore, since we can label inference rules by the corresponding connectives. We will use parameters instead of free variables, and therefore only deal with sentences.

DEFINITION 10 (sequent) A *sequent* is a set of sentences. Let S be a sequent and $\mathfrak{S}$ be a structure. We say that a sequent S is *is locally satisfied* in $\mathfrak{S}$ at a world $w \in \mathcal{W}$ *under a valuation* V if $\mathfrak{S}, w, V \Vdash A$ for all $A \in S$. A sequent S is *globally satisfied* in a structure $\mathfrak{S}$ under a valuation V if S is satisfied at every world of $\mathfrak{S}$ under V. A sequent S is called *locally (respectively, globally) satisfiable in* $\mathfrak{S}$ if it is locally (respectively, globally) satisfied in $\mathfrak{S}$ under some valuation. If S is locally satisfiable in $\mathfrak{S}$ we also say that $\mathfrak{S}$ is a *model* of S.

Thus, a sequent is understood as a (possibly infinite) conjunction of its members.

For a formula A and a set of formulas S we use A, S or S, A to denote the set $S \cup \{A\}$. Likewise, we write S_1, S_2 to denote the union of two sequents $S_1 \cup S_2$.

Sequent calculi for logics K, $\mathsf{K4}$, D, $\mathsf{D4}$, T, $\mathsf{S4}$ are shown in Figure 1.

DEFINITION 11 (inference, derivation, refutation) *The inference rules* of the sequent calculi are shown in Figure 1. We call an *inference* any particular instance of an inference rule. The *premises* of any inference or inference rule are the sequents above the bar, its *conclusion* is the sequent below the bar. An *axiom* is any conclusion of (ax). A *derivation* of a sequent S is a tree made of inferences and having S as the root. A derivation is called a *refutation* if all leaves in it are axioms.

We use the term *refutation* instead of a proof because the sequent calculi used in this paper establish unsatisfiability rather than validity.

EXAMPLE 12 Suppose that we wish to establish K-validity of the formula

$$\forall z([z]\forall x A(x) \supset \forall y[z]A(y)).$$

We turn this formula into its negation $\neg\forall z([z]\forall x A(x) \supset \forall y[z]A(y))$ and establish the unsatisfiability of the latter. To this end, we transform this formula into its negation normal form

$$\exists z([z]\forall x A(x) \wedge \exists y\langle z\rangle\neg A(y))$$

and try to find a refutation in the sequent calculus for K. An example refutation is as follows.

For all logics K, K4, D, D4, T, S4:

$$
\frac{}{S, A, \neg A}\ (\mathrm{ax}) \qquad \frac{S, A \quad S, B}{S, A \vee B}\ (\vee) \qquad \frac{S, A, B}{S, A \wedge B}\ (\wedge)
$$

$$
\frac{S, A(p)}{S, \exists x A(x)}\ (\exists) \qquad \frac{S, \forall x A(x), A(t)}{S, \forall x A(x)}\ (\forall) \qquad \frac{S^{[t]}, A}{S, \langle t \rangle A}\ (\langle t \rangle)
$$

For logics with seriality (D, D4):

$$
\frac{S^{[t]}}{S}\ ([t])
$$

For reflexive logics (T, S4):

$$
\frac{S, A}{S, [t]A}\ ([t])
$$

Logic $\mathcal{L}$	**Definition of $S^{[t]}$**
K, T, D	$S^{[t]} = \{A \mid [t]A \in S\}$
S4	$S^{[t]} = \{[t]A \mid [t]A \in S\}$
K4, D4	$S^{[t]} = \{A \mid [t]A \in S\} \cup \{[t]A \mid [t]A \in S\}$

The rule $(\exists)$ satisfies the *parameter condition*: p is a parameter having no occurrences in the conclusion of the rule.

Fig. 1. Sequent calculi

$$
\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{}{\forall x A(x), A(q), \neg A(q)}\ (\mathrm{ax})}{\forall x A(x), \neg A(q)}\ (\forall)}{[p]\forall x A(x), \langle p \rangle \neg A(q)}\ (\langle p \rangle)}{[p]\forall x A(x), \exists y \langle p \rangle \neg A(y)}\ (\exists)}{[p]\forall x A(x) \wedge \exists y \langle p \rangle \neg A(y)}\ (\wedge)}{\exists z([z]\forall x A(x) \wedge \exists y \langle z \rangle \neg A(y))}\ (\exists)
$$

This refutation is also a valid refutation in T and D.

To obtain a refutation in K4 and D4, we have to modify the top part of this refutation because of the difference in the definition of $S^{[t]}$:

$$
\cfrac{\cfrac{\cfrac{}{[p]\forall x A(x), \forall x A(x), A(q), \neg A(q)}\ (\mathrm{ax})}{[p]\forall x A(x), \forall x A(x), \neg A(q)}\ (\forall)}{[p]\forall x A(x), \langle p \rangle \neg A(q)}\ (\langle p \rangle)
$$

A refutation in S4 follows a different strategy because of the difference in the $([t])$ rule:

$$\cfrac{\cfrac{\cfrac{\cfrac{}{\forall x A(x), A(q), \neg A(q)}\ (\mathrm{ax})}{\forall x A(x), \neg A(q)}\ (\forall)}{[p]\forall x A(x), \neg A(q)}\ ([p])}{[p]\forall x A(x), \langle p\rangle \neg A(q)}\ (\langle p\rangle)$$

Since every formula has a negation normal form, we can restrict ourselves to negation normal forms.

We will augment the logics defined above with so-called *global assumptions*. Let Ψ be a set of sentences and $\mathcal{L}$ be one of the logics defined so far. We call a *sequent calculus for* $\mathcal{L}$ *with global assumptions* Ψ the calculus obtained from $\mathcal{L}$ by adding the rule $\dfrac{S, A}{S}\ (\Psi)$, where $A \in \Psi$.

Soundness and completeness of sequent calculi are proven in Fitting, Thalmann, and Voronkov [5]. The proofs are quite lengthy and not included. They are based on the method of Fitting [3]. Completeness is formulated as follows:

THEOREM 13 (Completeness of sequent calculi) *Let S be a set of sentences. If S has no refutation in the sequent calculus for $\mathcal{L}$ with the global assumptions Ψ, then there exists an $\mathcal{L}$-structure $\mathfrak{S}$ and a valuation V under which S is locally satisfied and all formulas in Ψ are globally satisfied.*

5 Tableau systems

Tableau systems formalize proof-search in sequent calculi. Tableaux are often introduced as trees of formulas, with inference rules on tableaux formulated in terms of *branches*. To simplify the presentation, we introduce tableaux as multisets of branches.

DEFINITION 14 (tableau, branch) A *tableau* is a finite multiset $S_1, S_2, \ldots$ of sequents, denoted $S_1 \mid S_2 \mid \cdots$. The *empty tableau* is denoted by $\#$. Every sequent S_i is called a *branch* of this tableau.

The tableau calculus for each logic studied in this paper can be obtained by a simple transformation of the corresponding sequent calculus. For every inference rule $\dfrac{S_1 \quad \ldots \quad S_n}{S}$ of the sequent calculus, the corresponding tableau rule has the form $\dfrac{S \mid \mathcal{T}}{S_1 \mid \ldots \mid S_n \mid \mathcal{T}}$ where $\mathcal{T}$ is any tableau. Note the reverse order of the sequents. The tableau calculus rules have the following intuitive meaning: suppose that we search for a refutation of S and all sequents in $\mathcal{T}$. Then, since there is a sequent calculus rule reducing S to the sequents $S_1, \ldots, S_n$, it is enough to find a refutation of $S_1, \ldots, S_n$ and all sequents in $\mathcal{T}$. To find a refutation for a formula A, we begin with a tableau consisting of one branch A and try to apply the tableau rules until no (unrefuted) branches remain.

Formally, the *tableau calculi* for $\mathcal{L}$ are shown in Figure 2.

For all logics $\mathsf{K}, \mathsf{K4}, \mathsf{D}, \mathsf{D4}, \mathsf{T}, \mathsf{S4}$:

$$\frac{S, A, \neg A \mid \mathcal{T}}{\mathcal{T}}\ |\mathrm{ax}| \qquad \frac{S, A \vee B \mid \mathcal{T}}{S, A \mid S, B \mid \mathcal{T}}\ |\vee| \qquad \frac{S, A \wedge B \mid \mathcal{T}}{S, A, B \mid \mathcal{T}}\ |\wedge|$$

$$\frac{S, \exists x A(x) \mid \mathcal{T}}{S, A(p) \mid \mathcal{T}}\ |\exists| \qquad \frac{S, \forall x A(x) \mid \mathcal{T}}{S, \forall x A(x), A(t) \mid \mathcal{T}}\ |\forall| \qquad \frac{S, \langle t \rangle A \mid \mathcal{T}}{S^{[t]}, A \mid \mathcal{T}}\ |\langle t \rangle|$$

For logics with seriality (D, $\mathsf{D4}$): For reflexive logics (T, $\mathsf{S4}$):

$$\frac{S \mid \mathcal{T}}{S^{[t]} \mid \mathcal{T}}\ |[t]| \qquad \frac{S, [t]A \mid \mathcal{T}}{S, A \mid \mathcal{T}}\ |[t]|$$

For logics with global assumptions Ψ:

$$\frac{S \mid \mathcal{T}}{S, A \mid \mathcal{T}}\ |\Psi|$$

Here $S^{[t]}$ is defined in the same way as for the sequent calculi. *Parameter condition*: in the rule $|\exists|$ p is a parameter having no occurrences in the premise of the rule. In the rule $|\Psi|$, $A \in \Psi$.

Fig. 2. Tableau calculi

THEOREM 15 (Equivalence of tableau calculi and sequent calculi) *A sequent S has a refutation in the sequent calculus for $\mathcal{L}$ (with global assumptions Ψ) if and only if there exists a derivation of # from S in the tableau calculus for $\mathcal{L}$ (with the global assumptions Ψ).* $\square$

6 Free-variable tableaux

In this section we change the tableau systems introduced in Section 5 into free-variable tableau systems. We will use the definitions introduced so far, except that we now allow free variables to occur in sequents and tableaux.

To avoid problems with the parameter condition in ($\forall$) rules, we introduce the *occurrence constraints* similar to those used in Voronkov [14], but we could use the "dynamic skolemization" technique introduced in Fitting [4] as well.

In this section we assume knowledge of the standard notions of substitutions and (idempotent, most general) unifiers see, e.g., Eder [1]. The application of a substitution σ to a term or formula E is denoted $E\sigma$. As usual, we may need to rename bound variables in a formula before we apply a substitution to it. Any idempotent most general unifier of n expressions $E_1, \ldots, E_n$ is denoted by $mgu(E_1, \ldots, E_n)$. The set of free variables of any expression E (e.g. formula or set of formulas) is denoted by $vars(E)$.

DEFINITION 16 (occurrence constraint) We call a *simple occurrence constraint* either $\bot$ or an expression $p \notin X$, where p is a parameter and X is a finite set of variables. An *occurrence constraint* is a conjunction of zero or more simple occurrence constraints. A conjunction of zero simple occurrence constraints is denoted by $\top$.

For any substitution σ and simple occurrence constraint $\mathcal{C} = (p \notin X)$, we denote by $\mathcal{C}\sigma$ the following simple occurrence constraint:

$$\mathcal{C}\sigma = \begin{cases} \bot, & \text{if } p \text{ occurs in } X\sigma; \\ p \notin vars(X\sigma), & \text{otherwise.} \end{cases}$$

When $\mathcal{C}$ is a conjunction $\mathcal{C}_1 \wedge \ldots \wedge \mathcal{C}_n$ of simple occurrence constraints, we denote by $\mathcal{C}\sigma$ the following occurrence constraint:

$$\mathcal{C}\sigma = \begin{cases} \bot, & \text{if } \mathcal{C}_i = \bot \text{ for some } i; \\ \mathcal{C}_1\sigma \wedge \ldots \wedge \mathcal{C}_n\sigma, & \text{otherwise.} \end{cases}$$

An occurrence constraint $\mathcal{C}$ is called *satisfiable* if $\mathcal{C}$ is not $\bot$. A *solution* to an occurrence constraint $\mathcal{C}$ is any substitution σ such that $\mathcal{C}\sigma \neq \bot$ and $x\sigma$ is ground for every variable occurring in $\mathcal{C}$. Evidently, an occurrence constraint $\mathcal{C}$ is satisfiable if and only if it has a solution: indeed, one can take as a solution any substitution mapping all variables of $\mathcal{C}$ into any ground term not containing parameters in $\mathcal{C}$.

We call a *constrained tableau* any pair consisting of a tableau $\mathcal{T}$ and constraint $\mathcal{C}$, denoted $\mathcal{T} \cdot \mathcal{C}$. Let $\mathcal{L}$ be one of the logics $\mathsf{K}, \mathsf{K4}, \mathsf{D}, \mathsf{D4}, \mathsf{T}$ and $\mathsf{S4}$. The *free-variable tableau calculi* for $\mathcal{L}$ are shown in Figure 3.

THEOREM 17 (Equivalence of free-tableau calculi and sequent calculi)
Let S be a set of sentences of the signature Σ. Then S has a refutation in the sequent calculus for $\mathcal{L}$ (with global assumptions Ψ) if and only if there exists a derivation of $\# \cdot \mathcal{C}$ from $S \cdot \top$ in the tableau calculus for $\mathcal{L}$ (with the global assumptions Ψ) such that $\mathcal{C}$ is satisfiable.

In order to prove this theorem, we will prove two results showing bisimulation between tableau derivations and free-variable tableau derivations.

Let $\mathcal{T} \cdot \mathcal{C}$ be a constrained tableau and σ be a substitution. We call the tableau $\mathcal{T}\sigma$ the *σ-instance* of $\mathcal{T} \cdot \mathcal{C}$ if $\mathcal{C}\sigma$ is satisfiable. A tableau $\mathcal{T}'$ is called an *instance* of $\mathcal{T} \cdot \mathcal{C}$ if it is a σ-instance of $\mathcal{T} \cdot \mathcal{C}$ for some σ.

LEMMA 18 *Suppose there exists a derivation of $\mathcal{T}_2 \cdot \mathcal{C}$ from $\mathcal{T}_1 \cdot \top$ in the free-variable tableau calculus for $\mathcal{L}$ with global assumptions Ψ. Then any instance of $\mathcal{T}_2 \cdot \mathcal{C}$ has a derivation from $\mathcal{T}_1$ in the tableau calculus for $\mathcal{L}$ with the global assumptions Ψ.*

PROOF. The proof is by induction on the length of derivations in the free-variable tableau calculus. When the derivation is of length 0, the claim is obvious, since $\mathcal{T}_1$ is the only instance of $\mathcal{T}_1 \cdot \top$, when $\mathcal{T}_1$ has no free variables. For derivations

$$\frac{S, A(\bar{s}), \neg A(\bar{t}) \mid \mathcal{T} \cdot \mathcal{C}}{\mathcal{T} mgu(\bar{s}, \bar{t}) \cdot \mathcal{C} mgu(\bar{s}, \bar{t})} \ |\mathrm{ax}| \qquad \frac{S, A \vee B \mid \mathcal{T} \cdot \mathcal{C}}{S, A \mid S, B \mid \mathcal{T} \cdot \mathcal{C}} \ |\vee| \qquad \frac{S, A \wedge B \mid \mathcal{T} \cdot \mathcal{C}}{S, A, B \mid \mathcal{T} \cdot \mathcal{C}} \ |\wedge|$$

$$\frac{S, \exists x A(x) \mid \mathcal{T} \cdot \mathcal{C}}{S, A(p) \mid \mathcal{T} \cdot \mathcal{C} \wedge p \notin vars(S, \exists x A(x))} \ |\exists| \qquad \frac{S, \forall x A(x) \mid \mathcal{T} \cdot \mathcal{C}}{S, \forall x A(x), A(y) \mid \mathcal{T} \cdot \mathcal{C}} \ |\forall|$$

$$\frac{S, \langle t \rangle A \mid \mathcal{T} \cdot \mathcal{C}}{S^{[t_1]}, \ldots, S^{[t_n]}, A \mid \mathcal{T}\sigma \cdot \mathcal{C}\sigma} \ |\langle t, t_1, \ldots, t_n \rangle|$$

For logics with seriality (D, D4):

$$\frac{S \mid \mathcal{T} \cdot \mathcal{C}}{S^{[t_1]}, \ldots, S^{[t_n]}, \mid \mathcal{T}\sigma \cdot \mathcal{C}\sigma} \ |[t_1, \ldots, t_n]|$$

For reflexive logics (T, S4):

$$\frac{S, [t]A \mid \mathcal{T} \cdot \mathcal{C}}{S, A \mid \mathcal{T} \cdot \mathcal{C}} \ |[t]|$$

For logics with global assumptions Ψ:

$$\frac{S \mid \mathcal{T} \cdot \mathcal{C}}{S, A \mid \mathcal{T} \cdot \mathcal{C}} \ |\Psi|$$

In the rule $|\langle t, t_1, \ldots, t_n \rangle|$, $\sigma = mgu(t, t_1, \ldots, t_n)$. In the rule $|[t_1, \ldots, t_n]|$, $\sigma = mgu(t_1, \ldots, t_n)$. In the rule $|\exists|$, p is new parameter, not occurring in the premise. In the rule $|\forall|$, y is a new variable, not occurring in the premise. In the rule $|\Psi|$, $A \in \Psi$.

Fig. 3. Free-variable tableau with constraints calculi

with at least one inference, consider the last inference of the derivation. We will consider only two cases, other cases are similar.

CASE: *the last inference is* $|\mathrm{ax}|$. Then it has the form $\dfrac{S, A(\bar{s}), \neg A(\bar{t}) \mid \mathcal{T} \cdot \mathcal{C}}{\mathcal{T}\sigma \cdot \mathcal{C}\sigma} \ |\mathrm{ax}|$, where $\sigma = mgu(\bar{s}, \bar{t})$.

Take any instance of $\mathcal{T}\sigma \cdot \mathcal{C}\sigma$, then this instance has the form $\mathcal{T}\sigma\tau$ for some substitution τ such that $\mathcal{C}\sigma\tau$ is satisfiable. We have to prove that $\mathcal{T}\sigma\tau$ is derivable from $\mathcal{T}_1$.

We claim that the following is a valid inference in the tableau calculus:

$$\frac{(S, A(\bar{s}), \neg A(\bar{t}) \mid \mathcal{T})\sigma\tau}{\mathcal{T}\sigma\tau} \ |\mathrm{ax}|. \tag{1}$$

Indeed, since σ is a unifier of $\bar{s}$ and $\bar{t}$, then $A(\bar{s})\sigma = A(\bar{t})\sigma$, hence $A(\bar{s})\sigma\tau = A(\bar{t})\sigma\tau$.

Since $\mathcal{C}\sigma\tau$ is satisfiable, $(S, A(\bar{s}), \neg A(\bar{t}) \mid \mathcal{T})\sigma\tau$ is a $\sigma\tau$-instance of $S, A(\bar{s}), \neg A(\bar{t}) \mid \mathcal{T} \cdot \mathcal{C}$. By the induction hypothesis, this instance has a derivation from $\mathcal{T}_1$. Add to this derivation the inference (1), then we obtain a required derivation of $\mathcal{T}\sigma\tau$.

CASE: *the last inference is* $|\exists|$.

$$\frac{S, \exists x A(x) \mid \mathcal{T} \cdot \mathcal{C}}{S, A(p) \mid \mathcal{T} \cdot \mathcal{C} \wedge p \notin vars(S, \exists x A(x))} \ (\exists).$$

Take any instance of $S, A(p) \mid \mathcal{T} \cdot \mathcal{C} \wedge p \notin vars(S, \exists x A(x))$, then there is a substitution τ such that this instance has the form $S\tau, A(p)\tau \mid \mathcal{T}\tau$ and $(\mathcal{C} \wedge p \notin vars(S, \exists x A(x)))\tau$ is satisfiable. We have to prove that $S\tau, A(p)\tau \mid \mathcal{T}\tau$ is derivable from $\mathcal{T}_1$.
By the definition of constraint satisfiability, since $(\mathcal{C} \wedge p \notin vars(S, \exists x A(x)))\tau$ is satisfiable, then $\mathcal{C}\tau$ is satisfiable and p does not occur in $S\tau, \exists x A(x)\tau$. Since p does not occur in $S\tau, \exists x A(x)\tau$, the following is a valid inference in the tableau calculus:

$$\frac{S\tau, \exists x A(x)\tau \mid \mathcal{T}\tau}{S\tau, A(p)\tau \mid \mathcal{T}\tau} \ (\exists). \tag{2}$$

By the induction hypothesis, every instance of $S, \exists x A(x) \mid \mathcal{T} \cdot \mathcal{C}$ has a derivation from $\mathcal{T}_1$. Since $\mathcal{C}\tau$ is satisfiable, we can take its τ-instance $S\tau, \exists x A(x)\tau \mid \mathcal{T}\tau$, this instance has a derivation from $\mathcal{T}_1$. Add to this derivation inference (2) and we obtain a required derivation of $S\tau, A(p)\tau \mid \mathcal{T}\tau$ from $\mathcal{T}_1$.

Now we want to prove a simulation result in the inverse direction. If we defined a sequent as a multiset of formulas, we could use an argument similar to the previous lemma. The use of sets instead of multisets causes some technical problems because the notion of instance does not work properly any more. To avoid these technical problems we give a definition of generalization that is nearly inverse to the notion of instance but takes into account some specific problems in the inverse simulation proof.

Let $\mathcal{T} = S_1 \mid \ldots \mid S_n$ and $\mathcal{T}' = S'_1 \mid \ldots \mid S'_n$ be two tableaux. We write $\mathcal{T} \sqsubseteq \mathcal{T}'$ if (i) for every $i = 1 \ldots n$ we have $S_i \subseteq S'_i$ and (ii) each parameter occurring in some $\mathcal{T}'$ also occurs in $\mathcal{T}$. Let $\mathcal{T} \cdot \mathcal{C}$ be a constrained tableau and $\mathcal{T}'$ a tableau. We call $\mathcal{T} \cdot \mathcal{C}$ a *σ-generalization* of $\mathcal{T}'$ if $\mathcal{T}' \sqsubseteq \mathcal{T}\sigma$ and $\mathcal{C}\sigma$ is satisfiable. We call $\mathcal{T} \cdot \mathcal{C}$ a *generalization* of $\mathcal{T}'$ if $\mathcal{T} \cdot \mathcal{C}$ is a σ-generalization of $\mathcal{T}'$ for some σ.

LEMMA 19 *Suppose there exists a derivation of $\mathcal{T}_2$ from $\mathcal{T}_1$ in the tableau calculus for $\mathcal{L}$ with global assumptions Ψ. Then some generalization of $\mathcal{T}_2$ has a derivation from $\mathcal{T}_1 \cdot \top$ in the free-variable tableau calculus for $\mathcal{L}$ with the global assumptions Ψ.*

PROOF. The proof is by induction on the length of derivations in the tableau calculus. When the derivation is of length 0, the claim is obvious, since $\mathcal{T}_1 \cdot \top$ is a generalization of $\mathcal{T}_1$. For derivations with at least one inference, consider the last inference of the derivation. We will consider only two cases, other cases are similar.

CASE: *the last inference is* $|\mathrm{ax}|$. Then it has the form $\dfrac{S, A, \neg A \mid \mathcal{T}}{\mathcal{T}}\ |\mathrm{ax}|$. By the induction hypothesis, some σ-generalization of $S, A, \neg A \mid \mathcal{T}$ is derivable from $\mathcal{T}_1 \cdot \top$. Then this generalization has a form $S', A', \neg B' \mid \mathcal{T}' \cdot \mathcal{C}$ such that $A'\sigma = A$, $B'\sigma = A$, $\mathcal{T} \sqsubseteq \mathcal{T}'\sigma$ and $\mathcal{C}\sigma$ is satisfiable. Then σ is a unifier of A' and B', therefore, there exists a most general unifier τ of A' and B' and a substitution δ such that $\tau\delta = \sigma$. Consider the following inference in the free-variable tableau calculus.

$$\frac{S', A', \neg B' \mid \mathcal{T}' \cdot \mathcal{C}}{\mathcal{T}'\tau \cdot \mathcal{C}\tau}\ |\mathrm{ax}|.$$

We claim that the conclusion of this inference is a generalization of $\mathcal{T}$, this will complete the proof of this case. To prove the claim, we have to find a substitution δ' such that (i) $\mathcal{T} \sqsubseteq \mathcal{T}'\tau\delta'$ and (ii) $\mathcal{C}\tau\delta'$ is satisfiable. Well, take δ' to be δ, then both (i) and (ii) follow from $\tau\delta' = \sigma$.

CASE: *the last inference is* $|\exists|$.

$$\frac{S, \exists x A(x) \mid \mathcal{T}}{S, A(p) \mid \mathcal{T}}\ |\exists| \tag{3}$$

By the induction hypothesis, some σ-generalization of $S, \exists x A(x) \mid \mathcal{T}$ is derivable from $\mathcal{T}_1 \cdot \top$. Then this generalization has a form $S', \exists x A'(x) \mid \mathcal{T}' \cdot \mathcal{C}$ such that (i) $S \subseteq S'\sigma$, (ii) every parameter occurring in $(S', \exists x A'(x))\sigma$ also occurs in $S, \exists x A(x) \mid \mathcal{T}$, (iii) $\exists x A'(x)\sigma = \exists x A(x)$, (iv) $\mathcal{T} \sqsubseteq \mathcal{T}'\sigma$, and (v) $\mathcal{C}\sigma$ is satisfiable. Consider the following inference in the free-variable tableau calculus.

$$\frac{S', \exists x A'(x) \mid \mathcal{T}' \cdot \mathcal{C}}{S', A'(p) \mid \mathcal{T}' \cdot \mathcal{C} \wedge p \notin \mathit{vars}(S', \exists x A'(x))}\ |\exists|.$$

Let us check that the parameter condition is satisfied. Suppose, by contradiction, that p occurs in $S', \exists x A'(x) \mid \mathcal{T}'$, then it also occurs in $(S', \exists x A'(x) \mid \mathcal{T}')\sigma$, hence also in $S, \exists x A(x) \mid \mathcal{T}$. This violates the parameter condition of (3).

We claim that the conclusion of this inference is a generalization of $S, A(p) \mid \mathcal{T}$, this will complete the proof of this case. We actually claim that the conclusion is the σ-generalization of $S, A(p) \mid \mathcal{T}$. All conditions on σ-generalization except for constraint satisfaction immediately follow from (i)–(iv) above. It remains to verify that $(\mathcal{C} \wedge p \notin \mathit{vars}(S', \exists x A'(x)))\sigma$ is satisfiable. $\mathcal{C}\sigma$ is satisfiable by (v) above, so it remains to check that p does not occur in $(S', \exists x A'(x))\sigma$. If p occurred in $(S', \exists x A'(x))\sigma$, then by (ii) above p would also occur in $S, \exists x A(x) \mid \mathcal{T}$, but this is impossible because of the parameter condition in (3).

Now we can prove soundness and completeness of the free-variable calculi.

1. Suppose S has a refutation in the sequent calculus for $\mathcal{L}$ with global assumptions Ψ. Then by Theorem 15 there exists a derivation of $\#$ from S in the tableau calculus for $\mathcal{L}$ with the global assumptions Ψ. Hence, by Lemma 19 there exists a derivation of some generalization of $\#$ from $S \cdot \top$ in the free-variable tableau calculus for $\mathcal{L}$. But any generalization of $\#$ has the form $\# \cdot \mathcal{C}$ for a satisfiable $\mathcal{C}$.

2. Suppose there exists a derivation of $\# \cdot \mathcal{C}$ from $S \cdot \top$ in the tableau calculus for $\mathcal{L}$ with the global assumptions Ψ such that $\mathcal{C}$ is satisfiable. By Lemma 18 any instance of $\# \cdot \mathcal{C}$ has a derivation from S in the tableau calculus for $\mathcal{L}$ with the global assumptions Ψ. Obviously, $\#$ is such an instance, so it is derivable from S as well. By Theorem 15 S has a refutation in the sequent calculus for $\mathcal{L}$ with the global assumptions Ψ.

7 Example refutation

Consider the following formula valid in term-modal K:

$$\forall x \exists y([y]P(y,y) \land [f(y)](P(f(y),f(y)) \supset P(y,f(y))) \supset [f(x)]P(x,f(x))).$$

For better readability, we will omit parenthesis in terms like $f(x)$ and write fx instead. We will establish the validity of this formula, i.e. unsatisfiability of its negation using the free-variable tableau calculus for K. First, we negate the formula and transform it into negation normal form:

$$\exists x \forall y([y]P(y,y) \land [fy](\neg P(fy,fy) \lor P(y,fy)) \land \langle fx \rangle \neg P(x,fx)),$$

and then show its refutation. The refutation is given in Figure 4. In the refutation we do not show the constraint, since it always has the form $p \notin \emptyset$ and is satisfiable. For better readability, we denote the inference steps by $\rightarrow$ followed by the name of the inference rule. We also group several similar inferences into one. For example, by $|\forall|^*$ we denoted a sequence of $|\forall|$ inferences, and by $|\land|^*$ a sequence of $|\land|$ inferences.

References

1. E. Eder. Properties of substitutions and unifications. *Journal of Symbolic Computations*, 1(1):31–48, 1985.
2. R. Fagin, J.Y. Halpern, Y. Moses, and M.Y. Vardi. *Reasoning about Knowledge.* The MIT Press, Cambridge, 1995.
3. M. Fitting. *Proof methods for modal and intuitionistic logics*, volume 169 of *Synthese Library.* Reidel Publ. Comp., 1983.
4. M. Fitting. First-order modal tableaux. *Journal of Automated Reasoning*, 4:191–213, 1988.
5. M. Fitting, L. Thalmann, and A. Voronkov. Term-modal logics. Technical Report UMCS-2000-6-4, Department of Computer Science, University of Manchester, January 2000.
6. J.W. Garson. Quantification in modal logic. In D. Gabbay and F. Guenther, editors, *Handbook in Philosophical Logic*, volume II, chapter II.5, pages 249–307. D. Reidel Publishing Company, 1984.
7. G. Gentzen. Untersuchungen über das logische Schließen. *Mathematical Zeitschrift*, 39:176–210, 405–431, 1934. Translated as [8].
8. G. Gentzen. Investigations into logical deduction. In M.E. Szabo, editor, *The Collected Papers of Gerhard Gentzen*, pages 68–131. North Holland, Amsterdam, 1969. Originally appeared as [7].

$$\exists x \forall y([y]P(y,y) \wedge [fy](\neg P(fy,fy) \vee P(y,fy)) \wedge \langle fx\rangle \neg P(x,fx)) \rightarrow \quad |\exists|$$

$$\forall y([y]P(y,y) \wedge [fy](\neg P(fy,fy) \vee P(y,fy)) \wedge \langle fp\rangle \neg P(p,fp)) \rightarrow \quad |\forall|^*$$

$$\begin{array}{l}\forall y([y]P(y,y) \wedge [fy](\neg P(fy,fy) \vee P(y,fy)) \wedge \langle fp\rangle \neg P(p,fp)),\\ [z]P(z,z) \wedge [fz](\neg P(fz,fz) \vee P(z,fz)) \wedge \langle fp\rangle \neg P(p,fp),\\ [u]P(u,u) \wedge [fu](\neg P(fu,fu) \vee P(u,fu)) \wedge \langle fp\rangle \neg P(p,fp) \rightarrow \quad |\wedge|^*\end{array}$$

$$\begin{array}{l}\forall y([y]P(y,y) \wedge [fy](\neg P(fy,fy) \vee P(y,fy)) \wedge \langle fp\rangle \neg P(p,fp)),\\ [z]P(z,z), \quad [fz](\neg P(fz,fz) \vee P(z,fz)), \quad \langle fp\rangle \neg P(p,fp),\\ [u]P(u,u), \quad [fu](\neg P(fu,fu) \vee P(u,fu)) \rightarrow \quad |\langle fp,z,fu\rangle|\end{array}$$

$$P(fp,fp), \quad \neg P(p,fp), \quad \neg P(fp,fp) \vee P(p,fp) \rightarrow \quad |\vee|$$

$$\begin{array}{l}P(fp,fp), \neg P(p,fp), \neg P(fp,fp) \ \ |\\ P(fp,fp), \neg P(p,fp), P(p,fp) \rightarrow \quad |\mathrm{ax}|^*\end{array}$$

$$\#$$

Fig. 4. Example refutation in the free-variable calculus

9. A.J. Grove. Naming and identity in epistemic logics part II: A first-order logic for naming. *Artificial Intelligence*, 74:311–350, 1995.
10. A.J. Grove and J.Y. Halpern. Naming and identity in a multi-agent epistemic logic. In J. Allen, R. Fikes, and E. Sandewall, editors, *KR'91. Proc. of the 2nd International Conference on Principles of Knowledge Representation and Reasoning*, pages 301–312, Cambridge, Massachusets, April 1991. Morgan Kaufmann.
11. J. Hintikka. *Knowledge and Belief.* Cornell University Press, Ithaca, New York, 1962.
12. K. Schütte. *Beweistheorie (in German).* Springer Verlag, 1960.
13. R.M. Smullyan. A unifying principle in quantification theory. In *Proc. Nat. Acad. Sci. U.S.A.*, volume 49, pages 828–832, 1963.
14. A. Voronkov. Proof search in intuitionistic logic based on constraint satisfaction. In P. Miglioli, U. Moscato, D. Mundici, and M. Ornaghi, editors, *Theorem Proving with Analytic Tableaux and Related Methods. 5th International Workshop, TABLEAUX '96*, volume 1071 of *Lecture Notes in Artificial Intelligence*, pages 312–329, Terrasini, Palermo Italy, May 1996.
15. L.A. Wallen. *Automated Deduction in Nonclassical Logics.* The MIT Press, 1990.