Axiomatic and Tableau-Based Reasoning for Kt(H,R)

Renate A. Schmidt [1]

School of Computer Science
University of Manchester, Manchester, UK

John G. Stell

School of Computing
University of Leeds, Leeds, UK

David Rydeheard

School of Computer Science
University of Manchester, Manchester, UK

Abstract

We introduce a tense logic, called $Kt(H, R)$, arising from logics for spatial reasoning. $Kt(H, R)$ is a multi-modal logic with two modalities and their converses defined with respect to a pre-order and a relation stable over this pre-order. We show $Kt(H, R)$ is decidable, it has the effective finite model property and reasoning in $Kt(H, R)$ is PSPACE-complete. Two complete Hilbert-style axiomatisations are given. The main focus of the paper is tableau-based reasoning. Our aim is to gain insight into the numerous possibilities of defining tableau calculi and their properties. We present several labelled tableau calculi for $Kt(H, R)$ in which the theory rules range from accommodating correspondence properties closely, to accommodating Hilbert axioms closely. The calculi provide the basis for decision procedures that have been implemented and tested on modal and intuitionistic problems.

1 Introduction

In this paper we consider a variety of different deduction approaches in the spectrum between the purely axiomatic approach and the explicitly semantic approach. Our investigation is focussed on a tense logic, called $Kt(H, R)$. $Kt(H, R)$ has forward and backward looking modal operators defined by two accessibility relations H and R. The frame conditions are reflexivity and transitivity of H, and *stability* of R with respect to H. The stability condition is

[1] Much of the work was conducted while visiting the Max-Planck-Institut für Informatik, Saarbrücken. Partial support from UK EPSRC research grant EP/H043748/1 is gratefully acknowledged.

defined as $H;R;H \subseteq R$, where ; denotes relational composition. This means in a Kripke frame, for any two states u and v, whenever there is an H-transition from u, followed by an R-transition and an H-transition, to v, then there is also an R-transition from u to v.

The logic $Kt(H, R)$ originates with recent work on a bi-intuitionistic tense logic, called BISKT, which is studied with the motivation to develop a theory of relations on graphs and applications to spatial reasoning [23]. Given an undirected graph G, we can consider Kripke frames where the set of states is the set of all edges and all nodes in G. On these states we make an H-transition from u to v when either $u = v$ or when u is an edge which is incident with the node v. The significance of relations R, which are stable with respect to H is that they correspond exactly to the union-preserving functions on the lattice of subgraphs of G. This justifies viewing these stable relations as 'relations on G'. One motivation for investigating relations on graphs comes from mathematical morphology as used in image processing [4]. In its basic form this uses relations on sets of pixels to generate operations that approximate images. These approximations are designed to emphasise significant features and to reduce other features (such as noise). Mathematical morphology on graphs is currently being developed in image processing [6], although without explicitly using relations on graphs. In a Kripke frame for BISKT, constructed from a graph, formulae are interpreted as subgraphs and the box and diamond modal operators arising from R are operations on subgraphs providing forms of the erosion and dilation operations in mathematical morphology. The precise relationship between these modal operators and the morphological operators described in [6] is still under investigation. Using the standard embedding of intuitionistic logics into modal logic, the logic BISKT can be embedded into $Kt(H, R)$ and properties such as decidability, the finite model property and complexity of $Kt(H, R)$ carry over to BISKT. Moreover, deduction methods for $Kt(H, R)$ and implementations can be used for BISKT.

$Kt(H, R)$ is of independent interest because the modal axiom(s) corresponding to the stability condition can be used to ascribe levels of awareness to agents in a multi-agent setting. The standard model for formalising knowledge and actions performed by agents, or events happening in an agent environment, uses the *S5*-modality as knowledge operator and K modalities as action operators. In $Kt(H, R)$, the $[H]$-modality and the $[R]$-modality can be seen as modelling knowledge and action operators.[2] $[H]\phi$ is read to mean 'the agent knows ϕ' and $[R]\phi$ is read to mean 'always after executing action R, ϕ holds'. The Axiom $S = [R]\phi \to [H][R][H]\phi$ corresponding to the stability condition $H ; R ; H \subseteq R$, can then be viewed as saying 'the agent knows that, after performing an action R, it knows the effects of the action'. Thus, it states the agent has (strong) awareness of performing action R and its effects.

[2] The formalisation is slightly more general, because the negative introspection axiom is not assumed for the $[H]$-modality but this is not critical because it can be easily added to the logic. Also, allowing multiple knowledge operators and multiple action operators does not pose any technical difficulties.

The logic $Kt(H,R)$ has an alternative axiomatisation in which the stability axiom S is equivalent to the two axioms $A = [R]\phi \to [H][R]\phi$ and $P = [R]\phi \to [R][H]\phi$. From an agent perspective, Axiom A says 'the agent knows, when action R is performed, then ϕ necessarily holds'; in other words, the agent is aware of action R. Axiom P says 'after performing action R the agent knows ϕ holds', i.e., it knows the post-condition has been realised. In some sense, Axioms A and P can be viewed as weak forms of no learning and perfect recall. No learning is typically formalised as $[R][H]\phi \to [H][R]\phi$, and perfect recall as $[H][R]\phi \to [R][H]\phi$ [26].

A contribution of this paper is a series of *labelled semantic tableau calculi*, also referred as *explicit tableau systems* [11], for the logic $Kt(H,R)$. Labelled semantic tableau systems are widely studied, cf. [13,8,5,7,22], and are related to labelled sequent and natural deduction systems, cf. [14,17,27]. Labelled semantic tableau systems are proof confluent, which means committing to an inference step never requires backtracking over the proof search for an unsatisfiable formula. Proof-confluent calculi provide more flexibility in designing and experimenting with search strategies, and they are easier to implement while preserving soundness and completeness. For the purposes of our theoretical and practical analyses and comparisons in this paper this is useful.

Labelled semantic tableau calculi of the pure semantic kind explicitly and directly construct Kripke models during the inference process. They use *structural rules* which are direct reflections of the background theory given by a set of characterising frame conditions. For example, for Axiom $4 = [H]\phi \to [H][H]\phi$ the structural rule is $H(s,t),\ H(t,u)\ /\ H(s,u)$ and ensures H will be a transitive relation. For logics with semantic characterisations, labelled tableau calculi using structural rules may be developed by systematic methods. A general method is described in [22,24].

Alternatively, the background theory can be accommodated as *propagation rules* [5]. The propagation rule for Axiom 4 is $s : [H]\phi,\ H(s,t)\ /\ t : [H]\phi$. Propagation rules accommodate the background theory not by representations of the correspondence properties, but by representations of inferences with the Hilbert axioms [18]. Propagation rules can be seen to attempt to speed up the inference process by not returning complete concrete models but only skeleton models and performing just enough inferences to determine both satisfiability and unsatisfiability.

In this paper we also explore the extreme case of basing the tableau rules of the background theory on direct representations of Hilbert axioms, e.g., using the rule $s : [H]\phi\ /\ s : [H][H]\phi$ for Axiom 4. This is an example of what we call an *axiomatic rule*. Calculi with such rules are seldom seen in the literature (but [14] is an exception), and some authors have suggested completeness and termination cannot be guaranteed with such rules. We show however complete and terminating tableau calculi based on such rules can be obtained.

After formally defining $Kt(H,R)$ in Section 2, we give two Hilbert-style axiomatisations in Section 3, which will form the basis for deriving various semantic labelled tableau calculi. Section 4 recalls standard notions of labelled tableau reasoning and presents a tableau calculus with structural rules, derived from the semantics of $Kt(H,R)$. With one of the Hilbert axiomatisations as a basis, a tableau calculus $Tab^{prop}_{A,P}$ using propagation rules is presented in Section 5. The underlying proof idea of the completeness of the calculus is the same as for the completeness of the axiomatic translation principle in [18]. A reduction of satisfiability problems in $Kt(H,R)$ to the guarded fragment, defined as a partial evaluation of the calculus $Tab^{prop}_{A,P}$, is presented in Section 6. This enables us to give decidability and complexity results for $Kt(H,R)$ and implies the effective finite model property. The various possibilities of mixing structural and propagation rules yield more sound, complete and terminating tableau calculi in Section 7. We also present sound, complete and terminating tableau calculi using axiomatic rules, including $s : [H]\phi \ / \ s : [H][H]\phi$. Implementations of the presented tableau calculi and experimental results are discussed in Section 8. The proofs may be found in the long version [19] of this paper.

2 The modal logic $Kt(H,R)$

$Kt(H,R)$ is an extension of a normal bi-modal logic with two pairs of tense operators. The connectives of $Kt(H,R)$ are those of propositional logic, we take as primitives the operators $\bot$, $\wedge$, $\neg$, as well as the four box operators $[H]$, $[R]$, $[\breve{H}]$ and $[\breve{R}]$. These are standard box operators interpreted over two relations H and R and their converses $\breve{H}$ and $\breve{R}$. Other Boolean operators including $\top$, $\vee$, $\rightarrow$ and the respective diamond operators can be defined as expected: $\top = \neg\bot$, $\phi \vee \psi = \neg(\neg\phi \wedge \neg\psi)$, $\phi \rightarrow \psi = \neg(\phi \wedge \neg\psi)$ and $\Diamond\phi = \neg\Box\neg\phi$ for each $\Diamond \in \{\langle H\rangle, \langle R\rangle, \langle\breve{H}\rangle, \langle\breve{R}\rangle\}$ and the corresponding $\Box \in \{[H], [R], [\breve{H}], [\breve{R}]\}$.

The semantics of $Kt(H,R)$ is defined over Kripke models of the form $\mathcal{M} = (W, H, R, \mathcal{V})$, where W is any non-empty set (the set of worlds), H and R are binary relations over W, and $\mathcal{V}$ is a valuation mapping defining where propositional variables hold. The semantics of formulae in $Kt(H,R)$ is inductively defined as follows.

$$\begin{array}{ll}
\mathcal{M}, w \Vdash p & \text{iff } w \in \mathcal{V}(p) \\
\mathcal{M}, w \nVdash \bot & \\
\mathcal{M}, w \Vdash \neg\phi & \text{iff } \mathcal{M}, w \nVdash \phi \\
\mathcal{M}, w \Vdash \phi \wedge \psi & \text{iff } \mathcal{M}, w \Vdash \phi \text{ and } \mathcal{M}, w \Vdash \psi \\
\mathcal{M}, w \Vdash [H]\phi & \text{iff } \mathcal{M}, v \Vdash \phi \text{ for all } H\text{-successors } v \text{ of } w \\
\mathcal{M}, w \Vdash [R]\phi & \text{iff } \mathcal{M}, v \Vdash \phi \text{ for all } R\text{-successors } v \text{ of } w \\
\mathcal{M}, w \Vdash [\breve{H}]\phi & \text{iff } \mathcal{M}, v \Vdash \phi \text{ for all } H\text{-predecessors } v \text{ of } w \\
\mathcal{M}, w \Vdash [\breve{R}]\phi & \text{iff } \mathcal{M}, v \Vdash \phi \text{ for all } R\text{-predecessors } v \text{ of } w
\end{array}$$

We further impose that

(i) H is reflexive and transitive, and

(ii) R is *stable with respect to* H, i.e., $H\,;R;H \subseteq R$, where ; denotes relational composition.

K	axiomatisation of propositional logic, modus ponens, axioms K and necessitation for all four modalities, and substitutivity		
$\breve{H}$	$\neg[H]\neg[\breve{H}]\phi \to \phi$	$\breve{\breve{H}}$	$\neg[\breve{H}]\neg[H]\phi \to \phi$
$\breve{R}$	$\neg[R]\neg[\breve{R}]\phi \to \phi$	$\breve{\breve{R}}$	$\neg[\breve{R}]\neg[R]\phi \to \phi$
T	$[H]\phi \to \phi$	$\breve{T}$	$[\breve{H}]\phi \to \phi$
4	$[H]\phi \to [H][H]\phi$	$\breve{4}$	$[\breve{H}]\phi \to [\breve{H}][\breve{H}]\phi$
S	$[R]\phi \to [H][R][H]\phi$	$\breve{S}$	$[\breve{R}]\phi \to [\breve{H}][\breve{R}][\breve{H}]\phi$

Table 1
Axiomatisation $\mathcal{H}_S$ of $Kt(H,R)$.

Ax $\mathcal{A}$	Frame conditions	
T	H is reflexive	$\forall x\, H(x,x)$
4	H is transitive	$\forall xyz\,(H(x,y) \wedge H(y,z) \to H(x,z))$
S	R is stable wrt. H	$\forall xyzu\,(H(x,y) \wedge R(y,z) \wedge H(z,u) \to R(x,u))$
A	R is ante-stable wrt. H	$\forall xyz\,(H(x,y) \wedge R(y,z) \to R(x,z))$
P	R is post-stable wrt. H	$\forall xyz\,(R(x,y) \wedge H(y,z) \to R(x,z))$

Table 2
Axioms and frame conditions.

As usual, $\mathcal{F} = (W, H, R)$ is referred to as the Kripke frame of $\mathcal{M}$. Any Kripke frame (W, H, R) for which (i) and (ii) hold is called a *$Kt(H,R)$-frame* and any model $(W, H, R, \mathcal{V})$ for which (i) and (ii) hold is called a *$Kt(H,R)$-model.* We refer to Kripke models (frames) defined over relations and their converses as *tense Kripke models (frames).*

It follows from results in Section 6 below that:

Theorem 2.1 *(i) $Kt(H,R)$ is decidable and has the effective finite model property.*

(ii) Satisfiability in $Kt(H,R)$ is PSPACE-complete.

3 Axiomatisation and alternative characterisation

Table 1 presents an axiomatisation $\mathcal{H}_S$ of $Kt(H,R)$. $\mathcal{H}_S$ is given as an extension of basic multi-modal logic with four modal operators. The axioms $\breve{H}$, $\breve{\breve{H}}$, $\breve{R}$ and $\breve{\breve{R}}$ define the pairs of tense operators. Axioms T and 4 define $[H]$ as an *S4*-modality. Similarly, $[\breve{H}]$ is defined as an *S4*-modality. S and $\breve{S}$ are the *stability axioms.* This means $Kt(H,R)$ is an extension of the basic tense logic Kt with two pairs of modalities: an *S4*-modality and a modality stable with respect to the *S4*-modality.

$Kt(H,R)$ is first-order definable, because the extra axioms are expressible by first-order conditions on tense Kripke frames: in particular, $\breve{H}$, $\breve{\breve{H}}$, $\breve{R}$ and $\breve{\breve{R}}$ by tautologies, and the remaining axioms have first-order correspondence properties, as given in Table 2. T and 4 means that H is a reflexive, transitive relation and the frame condition for S is (H,R)-stability, i.e., $H \,;\, R \,;\, H \subseteq R$.

K	axiomatisation of propositional logic, modus ponens, axioms K and necessitation for all four modalities, substitutivity		
$\breve{H}$	$\neg[H]\neg[\breve{H}]\phi \to \phi$	$\breve{\breve{H}}$	$\neg[\breve{H}]\neg[H]\phi \to \phi$
$\breve{R}$	$\neg[R]\neg[\breve{R}]\phi \to \phi$	$\breve{\breve{R}}$	$\neg[\breve{R}]\neg[R]\phi \to \phi$
T	$[H]\phi \to \phi$	$\breve{T}$	$[\breve{H}]\phi \to \phi$
4	$[H]\phi \to [H][H]\phi$	$\breve{4}$	$[\breve{H}]\phi \to [\breve{H}][\breve{H}]\phi$
A	$[R]\phi \to [H][R]\phi$	$\breve{A}$	$[\breve{R}]\phi \to [\breve{H}][\breve{R}]\phi$
P	$[R]\phi \to [R][H]\phi$	$\breve{P}$	$[\breve{R}]\phi \to [\breve{R}][\breve{H}]\phi$

Table 3
Axiomatisation $\mathcal{H}_{A,P}$ of $Kt(H,R)$.

(The frame conditions for the converse versions should be clear.)

Generalisations of Sahlqvist's correspondence and completeness results [25] give us:

Theorem 3.1 *The axiomatisation $\mathcal{H}_S$ of $Kt(H,R)$ is sound and complete with respect to the class of $Kt(H,R)$-frames.*

The following properties provide the basis for an alternative characterisation of $Kt(H,R)$-frames and models.

Lemma 3.2 *Let (W,H,R) be any relational structure where H is reflexive.*

(i) The following are equivalent:
a. R is stable with respect to H, i.e., $H\,;R\,;H \subseteq R$.
b. R is ante- and post-stable with respect to H, i.e., $H\,;R \subseteq R$ and $R\,;H \subseteq R$.

(ii) If R is H-stable, then $[R]$ is monotone with respect to H, i.e., for any $w,v \in W$, if $\mathcal{M},w \Vdash [R]\phi$ and $H(w,v)$, then $\mathcal{M},v \Vdash [R]\phi$.

Lemma 3.3 *In any $Kt(H,R)$-frame $\breve{R}$ has the same properties with respect to $\breve{H}$, as R has with respect to H. For example:*

(i) $\breve{H}$ is reflexive and transitive.

(ii) $\breve{R}$ is stable with respect to $\breve{H}$, i.e., $\breve{H}\,;\breve{R}\,;\breve{H} \subseteq \breve{R}$.

(iii) $\breve{R}$ is ante- and post-stable with respect to $\breve{H}$, i.e., $\breve{H}\,;\breve{R} \subseteq \breve{R}$. and $\breve{R}\,;\breve{H} \subseteq \breve{R}$.

(iv) $[\breve{R}]$ is monotone with respect to $\breve{H}$, i.e., for any $w,v \in W$, if $\mathcal{M},w \Vdash [\breve{R}]\phi$ and $H(v,w)$, then $\mathcal{M},v \Vdash [\breve{R}]\phi$.

These results imply $Kt(H,R)$-frames, in which (H,R)-stability holds, and tense (W,H,R)-frames, in which H is a pre-order and (H,R)-ante and post-stability hold, are equivalent.

Since ante- and post-stability of (H,R) are correspondence properties of the Axioms $A = [R]\phi \to [H][R]\phi$ and $P = [R]\phi \to [R][H]\phi$, an alternative axiomatisation of $Kt(H,R)$ is $\mathcal{H}_{A,P}$ as given in Table 3. In $\mathcal{H}_{A,P}$ the stability axioms S and $\breve{S}$ have been replaced by the axioms A, $\breve{A}$, P and $\breve{P}$.

Theorem 3.4 *(i) The axiomatisation $\mathcal{H}_{A,P}$ is sound and complete with respect to the class of tense (H,R)-frames, where H is a pre-order, and ante- and post-stability of (H,R) hold (cf. Table 2).*

(ii) $\mathcal{H}_{A,P}$ is equivalent to $\mathcal{H}_S$.

The axiomatisation $\mathcal{H}_{A,P}$ is the basis for the tableau calculus presented in Section 5 and the axiomatic translation presented in Section 6.

4 A semantic tableau calculus for $Kt(H,R)$

Tableau formulae in our calculi have one of the forms $\perp$, $s : \phi$, $H(s,t)$, $R(s,t)$, $s \approx t$ or $s \not\approx t$. s and t denote *labels* which are terms of a freely generated term algebra over a finite set of constants (denoted by $a, b, \ldots$) and four unary function symbols $f_{\neg\Box\phi}$, one for each modality $\Box \in \{[H], [R], [\breve{H}], [\breve{R}]\}$. $\approx$ is the equality symbol.

The semantics of tableau formulae is an appropriately defined extension of the semantics of modal formulae. The extension $(\mathcal{M}, \iota)$ of a $Kt(H,R)$-model $\mathcal{M}$ with an assignment ι mapping labels to worlds in W is called an *extended $Kt(H,R)$-model.* Satisfiability of tableau formulae in $(\mathcal{M}, \iota)$ is defined by:

$$
\begin{array}{llll}
\mathcal{M}, \iota \not\Vdash \perp & & \mathcal{M}, \iota \Vdash s : \phi & \text{iff } \mathcal{M}, \iota(s) \Vdash \phi \\
\mathcal{M}, \iota \Vdash H(s,t) & \text{iff } (\iota(s), \iota(t)) \in H & \mathcal{M}, \iota \Vdash R(s,t) & \text{iff } (\iota(s), \iota(t)) \in R \\
\mathcal{M}, \iota \Vdash s \approx t & \text{iff } \iota(s) = \iota(t) & \mathcal{M}, \iota \Vdash s \not\approx t & \text{iff } \iota(s) \neq \iota(t)
\end{array}
$$

Let Tab_S^{str} be the tableau calculus consisting of the *basic rules* and the *theory rules* given respectively in Figures 1 and 2. The basic rules are the standard decomposition rules for labelled modal formulae; as usual, there is one pair of rules for each primitive logical operator, plus the closure rule (cl). In the rules for negated box formulae we see how the function symbols are used to create new successors represented by Skolem terms. (Instead, new constants could be created, but an advantage of using Skolem terms is that no inference steps need to be recomputed when blocking occurs.) The theory rules are the reflexivity rule for H, the transitivity rule for H and the stability rule for (H,R). Since they are direct reflections of the frame conditions, following [5], they are referred to as *structural rules.*

A general form of blocking is provided by the *unrestricted blocking mechanism* [20,21], which is based on the use of the (ub) rule and an appropriate form of equality reasoning, for example, the equality rules in Figure 3. Adding the unrestricted blocking mechanism to a sound and complete labelled tableau calculus forces termination, when the logic has the (effective) finite model property. We denote the calculus extended with the unrestricted blocking mechanism by $Tab_S^{str}(ub)$.

The tableau inference process constructs derivation trees. Starting with a set of tableau formulae, the rules are applied in a top-down manner. This leads to the formulae being decomposed into smaller formulae. The application of

$$\text{(cl)}\ \frac{s:\phi,\ s:\neg\phi}{\bot} \qquad (\bot)\ \frac{s:\bot}{\bot} \qquad (\neg\neg)\ \frac{s:\neg\neg\phi}{s:\phi}$$

$$(\wedge)\ \frac{s:\phi\wedge\psi}{s:\phi,\ s:\psi} \qquad (\neg\wedge)\ \frac{s:(\neg\phi\wedge\psi)}{s:\neg\phi \mid s:\neg\psi}$$

$$([H])\ \frac{s:[H]\phi,\ H(s,t)}{t:\phi} \qquad (\neg[H])\ \frac{s:\neg[H]\phi}{H(s,f_{\neg[H]\phi}(s),\ f_{\neg[H]\phi}(s):\neg\phi}$$

$$([\breve{H}])\ \frac{s:[\breve{H}]\phi,\ H(t,s)}{t:\phi} \qquad (\neg[\breve{H}])\ \frac{s:\neg[\breve{H}]\phi}{H(f_{\neg[\breve{H}]\phi}(s),s),\ f_{\neg[\breve{H}]\phi}(s):\neg\phi}$$

$$([R])\ \frac{s:[R]\phi,\ R(s,t)}{t:\phi} \qquad (\neg[R])\ \frac{s:\neg[R]\phi}{R(s,f_{\neg[R]\phi}(s)),\ f_{\neg[R]\phi}(s):\neg\phi}$$

$$([\breve{R}])\ \frac{s:[\breve{R}]\phi,\ R(t,s)}{t:\phi} \qquad (\neg[\breve{R}])\ \frac{s:\neg[\breve{R}]\phi}{R(f_{\neg[\breve{R}]\phi}(s),s),\ f_{\neg[\breve{R}]\phi}(s):\phi}$$

Fig. 1. The basic tableau rules.

$$(T_c)\ \frac{}{H(s,s)} \qquad (4_c)\ \frac{H(s,t),\ H(t,u)}{H(s,u)} \qquad (S_c)\ \frac{H(s,t),\ R(t,u),\ H(u,v)}{R(s,v)}$$

Fig. 2. Structural theory rules of Tab_S^{str}.

Unrestricted blocking rule: $$\text{(ub)}\ \frac{}{s\approx t \mid s\not\approx t}$$

Paramodulation equality rules: $$\frac{s\not\approx s}{\bot} \qquad \frac{s\approx t}{t\approx s} \qquad \frac{s\approx t,\ G[s]_\lambda}{G[\lambda/t]}$$

Fig. 3. Unrestricted blocking and equality rules. G denotes any tableau formula. $G[s]_\lambda$ means s occurs as a subterm at position λ in G, and $G[\lambda/t]$ denotes the formula obtained by replacing s at position λ with t.

the $(\neg\wedge)$-rule splits the current tableau branch into two branches. As soon as $\bot$ is derived in a branch, the branch is regarded as *closed* and the expansion of this branch stops. The inference process continues with the extension of a not yet closed or not yet fully expanded branch. A branch is *open* when it is not closed. When, in an open branch, no more rules are applicable, the derivation stops because a model can be read off from the branch.

A tableau calculus is *sound* when for a satisfiable set of tableau formulae any fully expanded tableau derivation has an open branch. (A tableau derivation is fully expanded if all branches are either closed, or open and fully expanded.) A tableau calculus is *(refutationally) complete* if for any unsatisfiable set of tableau formulae there is a closed tableau derivation. Though we do not emphasise it or show it explicitly, the tableau calculi we present are in fact *constructively complete*, by which we mean for every fully expanded open branch a model of the input set exists (that can either be read off from the branch, or, for the tableau calculi using propagation rules, constructed from it). A tableau calculus is *terminating*, if any fully expanded tableau derivation has a finite open branch if the input set is satisfiable.

Theorem 4.1 *(i) The tableau calculus* Tab_S^{str} *is sound and complete.*

$$(T)\ \frac{s:[H]\phi}{s:\phi} \qquad (\breve{T})\ \frac{s:[\breve{H}]\phi}{s:\phi}$$

$$(4)\ \frac{s:[H]\phi,\ H(s,t)}{t:[H]\phi} \qquad (\breve{4})\ \frac{s:[\breve{H}]\phi,\ H(t,s)}{t:[\breve{H}]\phi)}$$

$$(A)\ \frac{s:[R]\phi,\ H(s,t)}{t:[R]\phi} \qquad (\breve{A})\ \frac{s:[\breve{R}]\phi,\ H(t,s)}{t:[\breve{R}]\phi}$$

$$(P)\ \frac{s:[R]\phi,\ R(s,t)}{t:[H]\phi} \qquad (\breve{P})\ \frac{s:[\breve{R}]\phi,\ R(t,s)}{t:[\breve{H}]\phi}$$

Fig. 4. Propagation theory rules of $Tab^{prop}_{A,P}$.

(ii) So is the extension $Tab^{str}_{S}(ub)$ *with unrestricted blocking. Moreover:*

(iii) $Tab^{str}_{S}(ub)$ *is terminating and provides a decision procedure for* $Kt(H,R)$.

The calculus Tab^{str}_{S} provides the baseline for the completeness proofs of the tableau calculi defined in the next two sections.

5 Using propagation rules

Applying the ideas of the axiomatic translation principle [18] to the axiomatisation $\mathcal{H}_{A,P}$, based on the ante- and post-stability axioms, produces the calculus $Tab^{prop}_{A,P}$ consisting of the basic rules in Figure 1 and the theory rules in Figure 4. The basic rules are the same as for the calculus in the previous section. They form the core also for the calculi defined in the next section. Only the theory rules are varied. In $Tab^{prop}_{A,P}$ the theory rules are propagation rules. Box formulae defined over H are propagated by the rules (4) and $(\breve{4})$ to H-successors and predecessors, while box formulae defined over R are propagated by the rules (A) and $(\breve{A})$ to H-successors and predecessors, and the rules (P) and $(\breve{P})$ propagate them over R-links but turn them into box formulae defined over H.

Proving soundness of the calculus is routine. The creative and more difficult part is proving completeness. Our proof uses a simulation argument in which we show every refutation in Tab^{str}_{S} can be mapped to a refutation in $Tab^{prop}_{A,P}$. For lack of space the proof appears only in the long version [19], but we note the proof gives useful insight into what the essential inference steps are, and has inspired the definition of the calculi in the next section.

Theorem 5.1 *The tableau calculus* $Tab^{prop}_{A,P}$ *is sound and complete.*

We refer to the left-most premises of any rule as the *main premises.* With two exceptions the modal formulae in the conclusions of all rules of $Tab^{prop}_{A,P}$ are subformulae of the main premise, or are negations of subformulae of the main premise. The exceptions are the rules (P) and $(\breve{P})$, which produce new $[H]\psi$ and $[\breve{H}]\psi$ formulae, but where ψ occurs in the input formula immediately below $[R]$ and $[\breve{R}]$ operators. This means indefinite formula growth does not occur. This observation is exploited in the proof of Theorem 6.1.

Because the unrestricted blocking rule is sound we can add the unrestricted blocking mechanism to the calculus while preserving soundness and complete-

$\mathcal{A}$	Axiom	Schema formulae $\mathrm{Ax}^{\mathcal{A}}(p)$	$\mathfrak{X}_{\mathcal{A}}$
T	$[H]p \to p$	$\forall x(\neg Q_{[H]p}(x) \vee Q_p(x))$	$\mathfrak{X}^{\epsilon}_{[H],\varphi} \cup \mathfrak{X}^{\epsilon}_{[R],\varphi}$
$\breve{T}$	$[\breve{H}]p \to p$	$\forall x(\neg Q_{[\breve{H}]p}(x) \vee Q_p(x))$	$\mathfrak{X}^{\epsilon}_{[H],\varphi} \cup \mathfrak{X}^{\epsilon}_{[R],\varphi}$
$\mathit{4}$	$[H]p \to [H][H]p$	$\forall x \forall y\,(\neg Q_{[H]p}(x) \vee \neg H(x,y) \vee Q_{[H]p}(y))$	$\mathfrak{X}^{\epsilon}_{[H],\varphi} \cup \mathfrak{X}^{\epsilon}_{[R],\varphi}$
$\breve{4}$	$[\breve{H}]p \to [\breve{H}][\breve{H}]p$	$\forall x \forall y\,(\neg Q_{[\breve{H}]p}(x) \vee \neg H(y,x) \vee Q_{[\breve{H}]p}(y))$	$\mathfrak{X}^{\epsilon}_{[\breve{H}],\varphi} \cup \mathfrak{X}^{\epsilon}_{[\breve{R}],\varphi}$
A	$[R]p \to [H][R]p$	$\forall x \forall y\,(\neg Q_{[R]p}(x) \vee \neg H(x,y) \vee Q_{[R]p}(y))$	$\mathfrak{X}^{\epsilon}_{[R],\varphi}$
$\breve{A}$	$[\breve{R}]p \to [\breve{H}][\breve{R}]p$	$\forall x \forall y\,(\neg Q_{[\breve{R}]p}(x) \vee \neg H(y,x) \vee Q_{[\breve{R}]p}(y))$	$\mathfrak{X}^{\epsilon}_{[\breve{R}],\varphi}$
P	$[R]p \to [R][H]p$	$\forall x \forall y\,(\neg Q_{[R]p}(x) \vee \neg R(x,y) \vee Q_{[H]p}(y))$	$\mathfrak{X}^{\epsilon}_{[R],\varphi}$
$\breve{P}$	$[\breve{R}]p \to [\breve{R}][\breve{H}]p$	$\forall x \forall y\,(\neg Q_{[\breve{R}]p}(x) \vee \neg R(y,x) \vee Q_{[\breve{H}]p}(y))$	$\mathfrak{X}^{\epsilon}_{[\breve{R}],\varphi}$

Table 4
Schema formulae.

ness. Termination is a consequence of the effective finite model property of $Kt(H,R)$ (shown below) and results in [21] (cf. also [22]).

Theorem 5.2 *(i) The extension* $\mathit{Tab}^{prop}_{A,P}(ub)$ *with unrestricted blocking is sound and complete.*

(ii) $\mathit{Tab}^{prop}_{A,P}(ub)$ *is terminating and provides a decision procedure for* $Kt(H,R)$.

6 Axiomatic translation

In this section we show the tableau calculus $\mathit{Tab}^{prop}_{A,P}$ of the previous section can serve as a basis for translating problems in $Kt(H,R)$ to the guarded fragment from which decidability and the finite model property of $Kt(H,R)$ then follow.

Let φ be any $Kt(H,R)$-formula. We assume φ is in a normal form using only the primitive operators of the logic. We define a mapping $\Pi^{\Delta}_{\mathfrak{X}}$ from $Kt(H,R)$-formulae to first-order formulae, called the *axiomatic translation* of $Kt(H,R)$. The definition follows the axiomatic translation principle in [18] and is in accordance with the tableau rules of $\mathit{Tab}^{prop}_{A,P}$ modulo one small variation. The variation is that the rule of double negation is worked into the definition.

The definition of $\Pi^{\Delta}_{\mathfrak{X}}$ is based on the axiomatisation $\mathcal{H}_{A,P}$, so we let Δ, which is the set of extra axioms, be the set of the axioms T, $\breve{T}$, 4, $\breve{4}$, A, $\breve{A}$, P and $\breve{P}$. $\mathfrak{X}$ is the set of instantiation sets for each extra axiom. Formally, $\mathfrak{X} = \{\mathfrak{X}_{\mathcal{A}}\}_{\mathcal{A}\in\Delta}$, where $\mathfrak{X}_{\mathcal{A}}$ is defined in the right-most column of Table 4. By definition, $\mathfrak{X}_{[\alpha],\varphi} = \{\psi \mid [\alpha]\psi \in \mathrm{Sf}(\varphi)\}$, where $\alpha \in \{H, R\}$, and $\mathrm{Sf}(\varphi)$ denotes the set of all subformulae of φ. This means that $\mathfrak{X}_{[\alpha],\varphi}$ is the set of subformulae occurring immediately below $[\alpha]$ in φ.

Now, let $\Pi^{\Delta}_{\mathfrak{X}}(\varphi)$ be the conjunction of (1)–(3).

$$\exists x\, Q_{\varphi}(x) \wedge \bigwedge\{\mathrm{Def}(\psi) \mid \psi \in \mathrm{Sf}(\varphi)\} \tag{1}$$

$$\bigwedge\{\mathrm{Ax}^{\mathcal{A}}(\psi) \mid \mathcal{A} \in \Delta,\ \psi \in \mathfrak{X}_{\mathcal{A}}\} \tag{2}$$

$$\bigwedge\{\mathrm{Def}(\psi) \mid \psi \in \mathrm{Sf}(X)\} \tag{3}$$

$$
\begin{array}{ccc}
\pi(\bot, x) = \bot \qquad \pi(p, x) = \top & & \pi(\neg p, x) = \neg Q_p(x) \\
\pi(\psi \wedge \phi, x) = Q_\psi(x) \wedge Q_\phi(x) & & \pi(\neg(\psi \wedge \phi), x) = Q_{\sim\psi}(x) \vee Q_{\sim\phi}(x) \\
\pi([H]\psi, x) = \forall y\, (H(x, y) \rightarrow Q_\psi(y)) & & \pi(\neg[H]\psi, x) = \exists y\, (H(x, y) \wedge Q_{\sim\psi}(y)) \\
\pi([\breve{H}]\psi, x) = \forall y\, (H(y, x) \rightarrow Q_\psi(y)) & & \pi(\neg[\breve{H}]\psi, x) = \exists y\, (H(y, x) \wedge Q_{\sim\psi}(y)) \\
\pi([R]\psi, x) = \forall y\, (R(x, y) \rightarrow Q_\psi(y)) & & \pi(\neg[R]\psi, x) = \exists y\, (R(x, y) \wedge Q_{\sim\psi}(y)) \\
\pi([\breve{R}]\psi, x) = \forall y\, (R(y, x) \rightarrow Q_\psi(y)) & & \pi(\neg[\breve{R}]\psi, x) = \exists y\, (R(y, x) \wedge Q_{\sim\psi}(y))
\end{array}
$$

Table 5
Definition of the basic translation mapping π.

$\mathrm{Def}(\psi)$ is defined by:

$$
\begin{aligned}
\mathrm{Def}(\psi) = {} & \forall x\, (Q_\psi(x) \rightarrow \pi(\psi, x)) \wedge \forall x\, (Q_\psi(x) \rightarrow \neg Q_{\sim\psi}(x)) \\
& \wedge \forall x\, (Q_{\sim\psi}(x) \rightarrow \pi(\sim\psi, x)).
\end{aligned}
$$

π is the basic translation mapping inductively defined in Table 5. Each unary predicate symbol Q_ψ represents the translation of modal formula ψ indicated in the index. Their purpose is to make the translation more effective through structure sharing (it is clear that further optimisations are possible). $\sim$ denotes complementation, i.e., $\sim\psi = \phi$ if $\psi = \neg\phi$, and $\sim\psi = \neg\psi$, otherwise. $\mathrm{Ax}^{\mathcal{A}}(\psi)$ in (2) is the conjunction of instances of all schema formulae $F\{p/\psi\}$ associated with each axiom $\mathcal{A}$. The schema formulae for $Kt(H, R)$ and the instantiation sets $\mathfrak{X}_{\mathcal{A}}$ for each axiom $\mathcal{A}$ are given in Figure 4. X in (3) is the set $\{[H]\psi \mid \psi \in \mathfrak{X}^{\epsilon}_{[R],\varphi}\} \cup \{[\breve{H}]\psi \mid \psi \in \mathfrak{X}^{\epsilon}_{[\breve{R}],\varphi}\}$. This concludes the definition of $\Pi^{\Delta}_{\mathfrak{X}}(\varphi)$.

Intuitively, $\Pi^{\Delta}_{\mathfrak{X}}(\varphi)$ is an encoding of the calculus $Tab^{prop}_{A,P}$ for a given formula φ. (1) and (3) are partial evaluations of applications of the basic rules, and (2) is the partial evaluation of applications of the theory propagation rules with respect to the instantiation sets for each axiom.

Theorem 6.1 *Let φ be any $Kt(H, R)$-formula. Then:*

(i) φ is satisfiable in $Kt(H, R)$ iff $\Pi^{\Delta}_{\mathfrak{X}}(\varphi)$ is first-order satisfiable.

(ii) $\Pi^{\Delta}_{\mathfrak{X}}(\varphi)$ can be computed in linear time and the size of $\Pi^{\Delta}_{\mathfrak{X}}(\varphi)$ is linear in the size of φ.

(iii) $\Pi^{\Delta}_{\mathfrak{X}}(\varphi)$ is equivalent to a guarded formula.

Thus, $\Pi^{\Delta}_{\mathfrak{X}}$ defines an effective translation of any $Kt(H, R)$-formula into the guarded fragment [1,12]. It defines, in fact, a mapping to the subfragment $GF1^-$ of the guarded fragment, which has been shown to be PSPACE-complete if the arity of predicates is finitely bounded [16]. Therefore, carrying over properties of the guarded fragment and $GF1^-$ give us decidability, the effective finite model property and complexity results for $Kt(H, R)$, as summarised in Theorem 2.1.

The guarded fragment can be decided by ordered resolution [10]; therefore, one further consequence is:

Theorem 6.2 *Both ordered resolution and ordered resolution with selection of binary literals as defined in [10] (see also [18]) decide the axiomatic translation*

$$(A_c)\ \frac{H(s,t),\ R(t,u)}{R(s,u)} \qquad\qquad (P_c)\ \frac{R(s,t),\ H(t,u)}{R(s,u)}$$

Fig. 5. Structural theory rules for ante- and post-stability.

of satisfiability problems in $Kt(H,R)$.

Because the definition incorporates the needed number of modal formula instantiations of the propagation rules, the axiomatic translation can be viewed and reformulated as an encoding in basic tense logic Kt with global satisfiability (and two tense operators) of the $Kt(H,R)$-satisfiability of a formula φ. This encoding can be viewed as a global reduction function in the sense of [15] for $Kt(H,R)$, however a crucial variation is the signature extension with propositional symbols corresponding to the Q_ψ symbols. This makes further manipulation more efficient [18].

The calculus Tab_S^{str} of Section 4 based on structural rules also provides a basis for a translation to first-order logic, namely, the standard (relational) translation of $Kt(H,R)$ with structural transformation. However, it is not a mapping to the guarded fragment or any other known solvable fragment of first-order logic.

7 Other terminating tableau calculi

As is already apparent from Sections 3–5 there are several quite different deduction approaches for the logic $Kt(H,R)$. Further possibilities involve tableau systems based on a mixture of structural and propagation rules. Replacing the propagation rules for the axioms T and $\breve{T}$ in $Tab_{A,P}^{prop}$ by the reflexivity rule (T_c) preserves soundness and completeness. The proof is a small adaptation of the proof of Theorem 5.1.

Theorem 7.1 *The calculus* $\mathrm{Tab}_{A,P}^{mix}$ *consisting of the basic tableau rules of Figure 1, the reflexivity rule* (T_c) *for* H *and the propagation rules* (4), $(\breve{4})$, (A), $(\breve{A})$, (P) *and* $(\breve{P})$ *for* 4, A *and* P *is sound and complete.*

Basing the rules on the frame conditions of the semantics of the alternative axiomatisation $\mathcal{H}_{A,P}$ is another (obvious) possibility:

Theorem 7.2 *The calculus* $\mathrm{Tab}_{A,P}^{str}$ *consisting of the basic tableau rules, the reflexivity and transitivity rules as well as the structural rules* (A_c) *and* (P_c) *for ante- and post-stability (see Figure 5) is sound and complete.*

Propagation rules can be viewed as partial expansions of the corresponding axioms, and the results of the previous two sections show these partial expansions are sufficient for completeness. It also means the way the axioms are used can be accordingly restricted. This is the idea underlying the next result.

Theorem 7.3 *The calculi* $\mathrm{Tab}_{A,P}^{ax}$ *and* Tab_S^{ax} *consisting of the basic tableau rules, the rules* (T) *and* $(\breve{T})$, *and the rules* (4^*), $(\breve{4}^*)$, (A^*), $(\breve{A}^*)$, (P^*) *and* $(\breve{P}^*)$, *respectively* (4^*), $(\breve{4}^*)$, (S^*) *and* $(\breve{S}^*)$ *(as in Figure 6), are sound and complete.*

$$
\begin{array}{llll}
(4^*) & \dfrac{s : [H]\phi}{s : [H]^*[H]\phi} & (\breve{4}^*) & \dfrac{s : [\breve{H}]\phi}{s : [\breve{H}]^*[\breve{H}]\phi)} \\[2ex]
(A^*) & \dfrac{s : [R]\phi}{s : [H]^*[R]\phi} & (\breve{A}^*) & \dfrac{s : [\breve{R}]\phi}{s : [\breve{H}]^*[\breve{R}]\phi} \\[2ex]
(P^*) & \dfrac{s : [R]\phi}{s : [R]^*[H]\phi} & (\breve{P}^*) & \dfrac{s : [\breve{R}]\phi}{s : [\breve{R}]^*[\breve{H}]\phi} \\[2ex]
(S^*) & \dfrac{s : [R]\phi}{s : [H][R]^*[H]\phi} & (\breve{S}^*) & \dfrac{s : [\breve{R}]\phi}{s : [\breve{H}][\breve{R}]^*[\breve{H}]\phi}
\end{array}
$$

Fig. 6. Axiomatic theory rules. * binds with the box operator preceding it.

The meaning of the marker * is that box formulae annotated with it are not expanded with any theory rules, only with the standard expansion rules, namely the standard box rules and the closure rule. Though the starred rules cause formulae to grow in size, the formula growth is only temporary because of the restriction. The restriction defines a refinement, which is immediate from the remark before the theorem and is explicit in the completeness proof (cf. [19]).

$\mathit{Tab}^{ax}_{A,P}$ and Tab^{ax}_{S} can be flexibly varied by using the structural rules or propagation rules for subsets of the axioms. When using propagation rules for stability some care is needed. We can show:

(i) The calculus consisting of the basic tableau rules and the following rules is sound and complete, where $Y = \{[H][R][H]\psi \mid [R]\psi \in \mathrm{Sf}(\varphi)\} \cup \{[\breve{H}][\breve{R}][\breve{H}]\psi \mid [\breve{R}]\psi \in \mathrm{Sf}(\varphi)\}$.

$$
\begin{array}{llll}
(\mathsf{cut}) & \dfrac{}{s : \phi \mid s : \neg\phi}\ \phi \in Y & (T_c) & \dfrac{}{H(s,s)} \\[2ex]
(4) & \dfrac{s : [H]\phi,\ H(s,t)}{t : [H]\phi} & (\breve{4}) & \dfrac{s : [\breve{H}]\phi,\ H(t,s)}{t : [\breve{H}]\phi)} \\[2ex]
(S) & \dfrac{s : [R]\phi,\ H(s,t),\ R(t,u)}{u : [H]\phi} & (\breve{S}) & \dfrac{s : [\breve{R}]\phi,\ H(t,s),\ R(u,t)}{u : [\breve{H}]\phi}
\end{array}
$$

(ii) The calculus as in (i) but with propagation rules for T is sound and complete.

In both cases omitting the cut rule leads to incompleteness. There is a connection between these calculi and the calculus Tab^{ax}_{S} of Theorem 7.3, from which it is clear that the $[R]$ and $[\breve{R}]$-formulae occurring in the cut formulae do not need to be expanded with the (S) or $(\breve{S})$-rules. For completeness, in fact, the rule $s : [R]\psi \ / \ s : [H][R]^*[H]\psi \mid s : \neg[H][R]^*[H]\psi$ and the converse version, with the same restrictions for the starred boxes, are sufficient. However, the right branch can always be almost immediately closed, so that no gain is apparent over the calculus Tab^{ax}_{S}. This was confirmed in experiments.

Each of the calculi in this section is terminating when endowed with the unrestricted blocking mechanism. Even the calculus $\mathit{Tab}^{ax}_{S}(ub)$ without the restrictions to starred box formulae can be shown to be sound, complete and terminating. An important assumption is that the rules are applied fairly, i.e., no non-redundant application of a rule is postponed indefinitely.

Finally we note that each of the presented tableau calculi provides the basis for a reduction to first-order logic and their soundness and completeness is a consequence of the soundness and completeness of the calculi, and the fact that derivations are defined over a bounded number of modal formulae. Then, in the case of the propagation and axiomatic rules corresponding, effective partial evaluations as in the axiomatic translation in Section 6 can be defined and proved sound and complete. The reductions for structural rules involve, as expected, the corresponding frame conditions.

8 Implementation and experiments

We implemented the tableau calculi by encoding them into first-order logic and using the Spass-yarralumla system. Spass-yarralumla is a bottom-up model generator based on the Spass theorem prover (Version 3.8d) [28]. Spass-yarralumla emulates the behaviour of semantic labelled tableau provers [2,3]. The resolution refinement used is ordered resolution and selection of at least one negative literal in every clause. The inference loop of Spass was slightly modified so that it always takes the least complex clause as the given clause, ground clauses with positive equality literals are eagerly split, and a branch with a positive equality literal is always explored first. Equality reasoning is realised by ordered forward and backward rewriting. Spass-yarralumla implements several blocking techniques. We used four forms: (i) sound ancestor blocking (i.e., blocking is applied to distinct terms s and t if one is a subterm of the other, flag `-bld`); (ii) unrestricted blocking as defined in Figure 3 (flag `-bld -ubl`); (iii) sound ancestor blocking on non-disjoint worlds (i.e., blocking is restricted to subterms on unary predicates, flag `-blu`); and (iv) sound anywhere blocking on non-disjoint worlds (flag `-blu -ubl`).

The encodings of the tableau calculi are implemented as an extension of the `ml2dfg` tool used for the empirical evaluation of the axiomatic translation principle in [18]. Because this earlier work was limited to the evaluation of extensions of basic modal logic K, we extended the implementation to handle multiple modalities and backward looking modalities, and we implemented the encodings of the structural, propagation and axiomatic tableau rules for Axioms S, A and P, and extended the implementation of the encodings for T and 4. Thirteen encodings were evaluated. These include encodings of $Tab^{str}_{A,P}$ and Tab^{str}_{S} based correspondence properties (named `KtAcPcTc4c` and `KtScTc4c` in the results tables), the encoding of the tableau calculus $Tab^{prop}_{A,P}$ using propagation rules which was implemented via the axiomatic translation as defined in Section 6 (`KtAPT4`), the encodings of $Tab^{ax}_{A,P}$ and Tab^{ax}_{S} (`KtA*P*T4*` and `KtS*T4*`) as well as mixes of encodings of correspondence properties, the axiomatic translation, and almost purely axiomatic encodings. All tested encodings are sound and complete.

Evaluations were performed on problems created for the investigation of the logic BISKT in [23], and modal logic problems consisting predominantly of problems used in the experiments of [18]. The BISKT problems include intuitionistic propositional logic and intuitionistic modal logic problems. The

Spass-yarralumla.												
Blocking	(i) `-bld`			(ii) `-bld -ubl`			(iii) `-blu`			(iv) `-blu -ubl`		
Encoding	S	U	S/U	S	U	S/U	S	U	S/U	S	U	S/U
`KtA*P*T4*`	9.1	8.7	17.8	15.1	76.7	91.8	18.8	16.9	35.6	15.0	12.2	27.2
`KtA*PT4*`	8.3	9.2	17.5	17.7	97.7	115.4	15.5	16.6	32.1	12.3	12.6	24.9
`KtAP*T4*`	8.0	8.5	16.5	16.7	75.0	91.7	15.8	15.9	31.8	11.3	12.4	23.8
`KtA*P*T4`	7.4	**6.7**	14.1	14.7	**44.5**	59.2	14.1	11.9	25.9	11.2	**9.6**	**20.9**
`KtA*PT4`	**6.7**	**6.4**	**13.2**	13.9	**42.0**	**55.9**	11.6	**11.0**	22.6	**10.4**	**9.8**	**20.2**
`KtAP*T4`	**6.5**	**6.6**	**13.0**	12.4	**42.6**	**55.1**	12.1	**11.4**	23.6	**9.7**	**9.6**	**19.3**
`KtAPT4`	**6.2**	**6.2**	**12.4**	11.7	**40.6**	**52.3**	**9.8**	**10.5**	**20.3**	**9.9**	**9.4**	**19.3**
`KtAPTc4`	**6.3**	7.2	**13.6**	**10.2**	58.5	68.7	**10.1**	**11.1**	**21.2**	**10.2**	11.8	21.9
`KtAcPcTc4c`	**6.4**	28.4	34.7	14.9	220.3	235.2	**9.5**	60.9	70.4	**9.5**	32.8	42.3
`KtS*T4*`	10.6	9.2	19.8	16.2	80.0	96.2	24.6	18.3	42.9	14.9	12.5	27.4
`KtS*T4`	7.9	**6.7**	14.6	13.1	45.1	58.2	16.1	**11.6**	27.7	**10.2**	**9.7**	**19.9**
`KtS*Tc4`	8.2	7.7	15.8	**11.2**	61.8	73.0	15.7	12.1	27.8	11.7	12.3	23.9
`KtScTc4c`	**6.5**	34.7	41.1	14.3	180.1	194.4	**9.8**	76.5	86.4	**9.8**	34.8	44.6

Table 6
Average running times in 10 ms. S = satisfiable, U = unsatisfiable,
S/U = satisfiable or unsatisfiable.

average size of the Spass files generated by `ml2dfg` varied between 5.4 KB and 5.5 KB for `KtScTc4c` and `KtAcPcTc4c` to 12.9 KB and 16.1 KB for `KtA*P*T4*` and `KtS*T4*`. This range is plausible because for structural rules the encoding is smallest and for the rules closest to axiom form the partial evaluation results in larger encodings. The input files were the same for the tests done with Spass-yarralumla and Spass in auto mode. In total there were 240 satisfiable and 150 unsatisfiable problems.

The tests were run on a Linux PC with a 3.30GHz Intel Core i3-2120 CPU and 10 GB RAM. Each problem was run with a timeout of 600 seconds. The problems and detailed results are available at `http://staff.cs.manchester.ac.uk/~schmidt/publications/kthr14/`.

Table 6 summarises the results obtained for runs with Spass-yarralumla. The best results in each column are highlighted in bold dark grey. To account for variability in measurement, results within 10% of the best values are highlighted in light grey. Looking at the table for Spass-yarralumla, on the whole, the encoding `KtAPT4` of propagation rules fared best for all forms of blocking tested. Similarly good results were obtained for the encodings `KtA*P*T4`, `KtA*PT4`, `KtAP*T4`, `KtAPTc4`, and to some extent `KtS*T4`. For satisfiable problems the encodings based on correspondence properties fared well, too, in two cases giving best results for `KtAcPcTc4c`. For unsatisfiable problems the performance was always significantly worse, especially for unrestricted blocking. In terms of blocking, for all encodings unrestricted blocking was most expensive on unsatisfiable problems. In contrast to other blocking techniques, unrestricted blocking generates models with domains of minimal size, which is a much harder problem than determining if models exist. Sound ancestor blocking produced best results for all encodings on both satisfiable and unsatisfiable problems.

Table 7 gives the results of Spass (Version 3.8d) [28] in auto mode. In auto

Spass in auto mode.				
Encoding	S	U	S/U	M
`KtA*P*T4*`	75.4(2)	5.3	80.7(2)	12.9
`KtA*PT4*`	120.8(2)	5.4	126.3(2)	12.6
`KtAP*T4*`	69.9(2)	5.3	75.2(2)	11.7
`KtA*P*T4`	102.5(1)	**4.8**	107.3(1)	9.8
`KtA*PT4`	**82.2**	**4.8**	**87.0**	9.6
`KtAP*T4`	32.2(2)	**4.7**	36.9(2)	8.6
`KtAPT4`	**80.3**	**4.9**	**85.2**	8.4
`KtAPTc4`	33.2(2)	**4.9**	38.1(2)	7.8
`KtAcPcTc4c`	4.6(131)	685.1(11)	689.7(142)	5.5
`KtS*T4*`	754.2(12)	5.8	760.1(12)	16.1
`KtS*T4`	669.3(6)	**4.9**	674.3(6)	11.6
`KtS*Tc4`	353.3(12)	**5.0**	358.4(12)	10.8
`KtScTc4c`	4.6(131)	54.7	59.3(142)	5.4

Table 7
Average running times in 10 ms. S = satisfiable, U = unsatisfiable, S/U = satisfiable or unsatisfiable, M = average size in KB of input files.

mode Spass used a form of ordered resolution with dynamic selection. The number of timeouts or unclean exits is indicated in brackets. For two encodings there were no timeouts: `KA*PT4` and `KAPT4`, and their performances were very close. Since Spass in auto mode is not a decision procedure for problems with chaining laws such as transitivity or the stability properties, the many timeouts for the encodings `KAcPcTc4c` and `KScTc4c` are no surprise. For all encodings, apart from these two, the performances were very close for unsatisfiable problems, with `KAP*T4` performing best. It is interesting how much faster these performances were than the best performances for Spass-yarralumla, but not unexpected. For unsatisfiable problems, tableau-like approaches need to construct a complete derivation tree in which every formula is grounded, and this is generally larger than the non-ground clause set derived with ordered resolution. For satisfiable problems, tableau approaches have an advantage because there is no need to explore the entire search space. Spass computes clause set completions, which are compact representations of all possible models, not just one model. With the axiomatic translation, back-translation is not a big obstacle [18], and the ability to compute entailments is useful.

The problems used in the evaluation can be divided into three groups. One group are problems of the logic BISKT from the investigation in [23]. Essentially these are intuitionistic propositional logic and intuitionistic modal logic problems that have been translated to $Kt(H,R)$. We also used the problems from the investigation of [18]. Because these are predominantly uni-modal problems we have used them as problems for the modality $[R]$ and the modality $[H]$ in separate runs.

Table 8 presents the experimental results differentiated by problem group. For the BISKT problems, results in very close proximity were shown for the encodings `KtA*P*T4`, `KtA*PT4`, `KtAP*T4`, `KtAPT4`, `KtAPTc4` and `KtS*T4`. Most best times were observed for `KtAPT4`. Sound ancestor blocking and unrestricted blocking produced best and worst results respectively. Interestingly we see

BISKT problems. Number of problems: 43 S, 103 U, 146 S/U.

Blocking	(i) `-bld`			(ii) `-bld -ubl`			(iii) `-blu`			(iv) `-blu -ubl`		
Encoding	S	U	S/U	S	U	S/U	S	U	S/U	S	U	S/U
`KtA*P*T4*`	4.2	10.1	14.3	**4.1**	84.4	88.6	5.8	19.9	25.7	4.2	13.2	17.4
`KtA*PT4*`	4.1	10.5	14.5	**4.1**	119.4	123.5	5.8	20.4	26.2	4.2	14.0	18.2
`KtAP*T4*`	4.0	9.3	13.3	4.2	84.0	88.2	5.7	19.2	24.9	4.2	13.0	17.2
`KtA*P*T4`	**3.5**	**7.0**	**10.5**	**3.8**	**61.5**	**65.3**	**4.6**	12.8	17.4	**3.8**	**10.9**	**14.7**
`KtA*PT4`	**3.7**	**7.0**	**10.6**	**3.9**	**59.9**	**63.8**	**4.5**	**12.2**	**16.6**	**3.7**	**11.2**	**14.9**
`KtAP*T4`	**3.4**	**6.9**	**10.3**	**3.7**	**58.6**	**62.3**	**4.4**	**12.0**	**16.4**	**4.0**	**10.8**	**14.8**
`KtAPT4`	**3.5**	**6.6**	**10.1**	**3.8**	**57.0**	**60.8**	**4.4**	**11.2**	**15.6**	**3.7**	**10.4**	**14.1**
`KtAPTc4`	**3.6**	8.0	11.6	**3.8**	85.6	89.4	**4.4**	**12.1**	**16.5**	**3.8**	14.1	17.9
`KtAcPcTc4c`	**3.6**	19.1	22.7	4.2	323.1	327.3	**4.3**	43.1	47.3	**3.7**	34.7	38.4
`KtS*T4*`	4.3	10.1	14.4	**4.1**	90.6	94.7	6.0	22.1	28.1	4.3	13.8	18.0
`KtS*T4`	**3.7**	**7.2**	**10.9**	**4.0**	**62.4**	**66.4**	4.9	13.0	17.8	**4.0**	**11.1**	**15.1**
`KtS*Tc4`	**3.7**	8.6	12.3	**4.0**	89.3	93.3	**4.6**	14.6	19.2	**3.8**	15.3	19.1
`KtScTc4c`	**3.5**	28.3	31.9	4.3	261.3	265.6	**4.4**	64.9	69.3	**4.0**	37.6	41.6

Modal $[R]$ **problems.** Number of problems: 118 S, 4 U, 122 S/U.

Blocking	(i) `-bld`			(ii) `-bld -ubl`			(iii) `-blu`			(iv) `-blu -ubl`		
Encoding	S	U	S/U	S	U	S/U	S	U	S/U	S	U	S/U
`KtA*P*T4*`	11.4	4.8	16.1	15.9	5.0	20.9	28.3	10.0	38.3	19.7	5.8	25.4
`KtA*PT4*`	10.1	6.0	16.1	20.2	4.8	25.0	21.0	9.8	30.7	13.9	5.5	19.4
`KtA*P*T4`	9.5	4.5	14.0	18.7	5.0	23.7	21.7	10.0	31.7	14.6	5.8	20.3
`KtAP*T4*`	9.2	5.2	14.5	18.1	5.0	23.1	22.4	10.0	32.4	11.5	5.8	**17.3**
`KtA*PT4`	8.4	4.5	12.9	17.5	4.8	22.3	16.3	9.2	25.6	12.4	5.8	18.1
`KtAP*T4`	7.8	4.5	12.3	14.9	4.8	19.7	17.1	9.2	26.3	10.7	5.5	**16.2**
`KtAPT4`	7.3	4.2	11.5	12.8	4.5	17.3	12.8	9.2	22.1	11.4	5.8	**17.1**
`KtAPTc4`	7.8	4.5	12.3	**11.1**	4.5	**15.6**	13.0	9.0	22.0	11.0	5.5	**16.5**
`KtAcPcTc4c`	**5.6**	4.5	**10.1**	15.8	4.8	20.5	**8.5**	9.0	**17.5**	**9.6**	6.2	**15.9**
`KtS*T4*`	14.9	5.0	19.9	18.1	5.5	23.6	39.7	10.5	50.2	19.7	7.2	27.0
`KtS*T4`	10.8	4.8	15.5	14.7	5.0	19.7	25.5	9.8	35.3	11.8	6.2	18.1
`KtS*Tc4`	11.1	4.5	15.6	14.2	5.0	19.2	24.7	10.8	35.4	14.5	6.2	20.8
`KtScTc4c`	**5.7**	4.5	**10.2**	14.1	4.8	18.9	**8.6**	9.5	**18.1**	10.9	5.8	**16.7**

Modal $[H]$ **problems.** Number of problems: 79 S, 43 U, 122 S/U.

Blocking	(i) `-bld`			(ii) `-bld -ubl`			(iii) `-blu`			(iv) `-blu -ubl`		
Encoding	S	U	S/U	S	U	S/U	S	U	S/U	S	U	S/U
`KtA*P*T4*`	9.5	8.7	18.3	22.1	81.0	103.1	14.5	12.8	27.4	16.8	13.2	30.0
`KtA*PT4*`	9.6	9.0	18.5	22.0	80.7	102.7	14.7	**12.5**	27.2	17.6	13.0	30.6
`KtAP*T4*`	9.8	8.9	18.6	23.5	81.6	105.0	14.6	12.7	27.4	17.0	12.9	29.9
`KtA*P*T4`	**7.2**	**6.9**	**14.1**	16.7	**15.9**	**32.6**	**10.5**	**11.5**	**22.0**	**13.4**	**10.1**	**23.4**
`KtA*PT4`	**7.0**	**7.0**	**14.0**	15.7	**15.8**	**31.5**	**10.3**	**11.7**	**22.0**	**13.3**	**9.5**	**22.9**
`KtAP*T4`	**7.1**	**6.9**	**14.0**	15.6	**16.2**	**31.9**	**10.2**	**11.4**	**21.6**	**13.2**	**9.5**	**22.7**
`KtAPT4`	**7.0**	**6.8**	**13.9**	15.6	**15.8**	**31.3**	**10.2**	**11.5**	**21.8**	**13.2**	**9.7**	**22.8**
`KtAPTc4`	**7.0**	**7.1**	**14.1**	**13.4**	**16.7**	**30.1**	**10.0**	**11.7**	**21.7**	15.3	**10.0**	25.3
`KtAcPcTc4c`	9.8	58.1	67.9	21.5	47.4	68.9	16.4	121.4	137.8	**14.2**	38.6	52.8
`KtS*T4*`	9.5	8.7	18.3	22.1	81.2	103.3	14.6	13.0	27.6	17.1	13.0	30.2
`KtS*T4`	**7.2**	**6.8**	**13.9**	16.3	**17.1**	33.5	**10.2**	**11.9**	**22.1**	**13.2**	**9.7**	**22.9**
`KtS*Tc4`	**7.2**	**7.4**	**14.6**	**13.2**	**16.9**	**30.1**	**10.2**	**11.5**	**21.6**	**14.3**	**10.0**	**24.3**
`KtScTc4c`	10.4	61.6	71.9	21.7	50.8	72.5	17.0	128.9	145.9	**13.3**	42.9	56.2

Table 8
Average running times in 10 ms for the different problem sets. S = satisfiable, U = unsatisfiable, S/U = satisfiable or unsatisfiable.

that for satisfiable problems the performances were very close for all blocking techniques.

For the modal problems with mainly one modality, the $[R]$ modality, the sample of unsatisfiable problems is very small so we focus just on the results for satisfiable problems. Here, the standard translation or structural rules gave best results, except for the case of `-bld -ubl` and `-blu -ubl`. That the structural rules showed better performance is explained by the fact that the problems contain no $[H]$ modalities, only $[R]$ modalities. This means while especially the starred rules are applicable, the rules (A_c), (P_c) and (S_c) are not. Investigation of the results for unrestricted blocking has revealed that for `KtAcPcTc4c` the results are typically better than for `KtAPTc4`, except for one particular large and difficult problem where `KtAcPcTc4c` was about two times slower than `KtAPTc4` (1378 ms as opposed to 711 ms), which has affected the average results.

The results for the modal $[H]$ problems are interesting because although good performances are again obtained for `KtA*P*T4`, `KtA*PT4`, `KtAP*T4`, `KtAPT4` and `KtS*T4`, good performances in the same range were also obtained for `KtAPTc4` and `KtS*Tc4`. Here the systems based on the starred rules for A, P and S fared very well. Since the problems contain no $[R]$ modalities, these rules are not applicable resulting in fewer inference steps.

Overall, the results confirm that different performances should be expected for different methods on problem classes with different characteristics.

The reasons for implementing the tableau calculi as described were twofold. First, to get insight into the relative performances of different approaches and the properties of different techniques, we wanted a *fair* comparison. Second, using the `ml2dfg` tool and SPASS-YARRALUMLA was an easy way to test different sets of tableau rules and different rule refinements. That models can be read off from the output, aided quick discovery of less effective rule sets and counter-examples for incomplete rule sets, which was extremely useful during the development process. SPASS allowed us to confirm answers with a completely different approach.

9 Conclusion

We have introduced a tense logic $Kt(H,R)$ with two modalities interacting in a non-trivial way. We defined a range of different tableau calculi emerging in a systematic way from axiomatisations and the semantics of the logic. Via effective encodings these calculi can be mapped in various ways to the guarded fragment. This means any decision procedure for the guarded fragment can be used as decision procedure for $Kt(H,R)$. The results of the experiments with implementations of the tableau calculi with SPASS-YARRALUMLA, and SPASS using ordered resolution, give useful insights into the practical properties and relative efficiency of the different deduction approaches.

A more comprehensive empirical investigation needs to be done, but already several observations can be made. First, there are many more ways of deciding modal logics than is usually assumed. Second, we have gained useful insight into how and to what extent different approaches fit together and map to each other. Third, the behaviour of procedures depends a lot on the inference rules (or the transformations in the encodings), rule refinements, termination techniques, and what kind of deduction approaches are used. Fourth, a detailed analysis of the results revealed different performances can be observed for different approaches on problems with different properties (e.g., problems that are predominantly satisfiable, or are predominantly unsatisfiable, or have one of modalities dominate).

Though the focus has been on $Kt(H, R)$, the techniques and ideas presented in this paper are of general nature and provide a useful methodology for developing practical decision procedures for modal logics. Some aspects are completely routine. In particular, the structural rules can be obtained from the Hilbert axioms using methods of automated correspondence theory (cf. [9]) and tableau synthesis [22,24], and soundness and completeness of the calculi Tab_S^{str}, $Tab_S^{str}(ub)$, $Tab_{A,P}^{str}$ and $Tab_{A,P}^{str}(ub)$ are easily obtained. The main aspect for which creativity is required and is specific to $Kt(H, R)$ is the development of the tableau calculi based on propagation or axiomatic rules and the axiomatic translation to the guarded fragment. Here the contribution of the paper has been to extend the ideas of the axiomatic translation principle from [18]. Key is finding effective refinements and showing completeness and termination, which is in general non-trivial and will not always be possible. This gave us also the effective finite model property and termination of the presented tableau systems via the results in [21] and unrestricted blocking.

All in all, because of the ubiquity of modal logics, we believe this kind of systematic research of decidability, proof theory, refinements and relative efficiency is widely applicable and useful, and should be extended to more logics, more types of tableau approaches, other deduction approaches, different provers and more problem sets.

References

[1] Andréka, H., I. Németi and J. van Benthem, *Modal languages and bounded fragments of predicate logic*, J. Philos. Logic **27** (1998), pp. 217–274.

[2] Baumgartner, P. and R. A. Schmidt, *Blocking and other enhancements for bottom-up model generation methods*, in: *Proc. IJCAR 2006*, LNAI **4130** (2006), pp. 125–139.

[3] Baumgartner, P. and R. A. Schmidt, *Blocking and other enhancements for bottom-up model generation methods* (2008), manuscript.

[4] Bloch, I., H. J. A. M. Heijmans and C. Ronse, *Mathematical morphology*, in: M. Aiello, I. Pratt-Hartmann and J. van Benthem, editors, *Handbook of Spatial Logics*, Springer, 2007 pp. 857–944.

[5] Castilho, M. A., L. Fariñas del Cerro, O. Gasquet and A. Herzig, *Modal tableaux with propagation rules and structural rules*, Fund. Inform. **3–4** (1997), pp. 281–297.

[6] Cousty, J., L. Najman, F. Dias and J. Serra, *Morphological filtering on graphs*, Computer Vision and Image Understanding **117** (2013), pp. 370–385.

[7] Fariñas del Cerro, L. and O. Gasquet, *A general framework for pattern-driven modal tableaux*, Logic J. IGPL **10** (2002), pp. 51–83.

[8] Fitting, M., *Tableau methods of proof for modal logics*, Notre Dame J. Formal Logic **13** (1972), pp. 237–247.

[9] Gabbay, D. M., R. A. Schmidt and A. Szałas, "Second-Order Quantifier Elimination: Foundations, Computational Aspects and Applications," College Publ., 2008.

[10] Ganzinger, H. and H. de Nivelle, *A superposition decision procedure for the guarded fragment with equality*, in: *Proc. LICS 1999* (1999), pp. 295–303.

[11] Goré, R., *Tableau methods for modal and temporal logics*, in: M. D'Agostino, D. Gabbay, R. Hähnle and J. Posegga, editors, *Handbook of Tableau Methods*, Kluwer, 1999 pp. 297–396.

[12] Grädel, E., *On the restraining power of guards*, J. Symbolic Logic **64** (1999), pp. 1719–1742.

[13] Hughes, G. E. and M. J. Cresswell, "An Introduction to Modal Logic," Routledge, London, 1968.

[14] Indrzejczak, A., "Natural Deduction, Hybrid Systems and Modal Logics," Trends in Logic **30**, Springer, 2010.

[15] Kracht, M., "Tools and Techniques in Modal Logic," Elsevier, 1999.

[16] Lutz, C., U. Sattler and S. Tobies, *A suggestion of an n-ary description logic*, in: *Proc. DL 1999* (1999), pp. 81–85.

[17] Negri, S., *On the duality of proofs and countermodels in labelled sequent calculi*, in: *Proc. TABLEAUX 2013*, LNCS **8123**, Springer, 2013 pp. 5–9.

[18] Schmidt, R. A. and U. Hustadt, *The axiomatic translation principle for modal logic*, ACM Trans. Comput. Log. **8** (2007), pp. 1–55.

[19] Schmidt, R. A., J. G. Stell and D. Rydeheard, *Axiomatic and tableau-based reasoning for Kt(H,R)* (2014), available from `http://www.cs.man.ac.uk/~schmidt/publications/SchmidtStellRydeheard14b.html`.

[20] Schmidt, R. A. and D. Tishkovsky, *Using tableau to decide expressive description logics with role negation*, in: *Proc. ISWC 2007 + ASWC 2007*, LNCS **4825** (2007), pp. 438–451.

[21] Schmidt, R. A. and D. Tishkovsky, *A general tableau method for deciding description logics, modal logics and related first-order fragments*, in: *Proc. IJCAR 2008*, LNCS **5195** (2008), pp. 194–209.

[22] Schmidt, R. A. and D. Tishkovsky, *Automated synthesis of tableau calculi*, Logical Methods in Comput. Sci. **7** (2011), pp. 1–32.

[23] Stell, J. G., R. A. Schmidt and D. Rydeheard, *Tableau development for a bi-intuitionistic tense logic*, in: *Proc. RAMiCS 14*, LNCS **8428** (2014), pp. 412–428.

[24] Tishkovsky, D. and R. A. Schmidt, *Refinement in the tableau synthesis framework*, arXiv e-Print 1305.3131v1 (2013).

[25] van Benthem, J., *Correspondence theory*, in: D. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic*, Reidel, 1984 pp. 167–247.

[26] van der Hoek, W., *Logical foundations of agent-based computing*, in: *Muti-Agent Systems and Applications*, LNAI **2086**, Springer, 2001 pp. 50–73.

[27] Viganò, L., "Labelled Non-Classical Logics," Kluwer, 2000.

[28] Weidenbach, C., R. A. Schmidt, T. Hillenbrand, R. Rusev and D. Topic, *System description: SPASS version 3.0*, in: *Proc. CADE-21*, LNAI **4603** (2007), pp. 514–520.