

# Constructive interpolation in hybrid logic

Patrick Blackburn                      Maarten Marx\*  
INRIA, Lorraine                      Universiteit van Amsterdam  
patrick@aplog.org                      marx@science.uva.nl

September 30, 2002

## Abstract

Craig's interpolation lemma (if  $\varphi \rightarrow \psi$  is valid, then  $\varphi \rightarrow \theta$  and  $\theta \rightarrow \psi$  are valid, for  $\theta$  a formula constructed using only primitive symbols which occur both in  $\varphi$  and  $\psi$ ) fails for many propositional and first order modal logics. The interpolation property is often regarded as a sign of well-matched syntax and semantics. Hybrid logicians claim that modal logic is missing important syntactic machinery, namely tools for referring to worlds, and that adding such machinery solves many technical problems. The paper presents strong evidence for this claim by defining interpolation algorithms for both propositional and first order hybrid logic. These algorithms produce interpolants for the hybrid logic of every elementary class of frames satisfying the property that a frame is in the class if and only if all its point-generated subframes are in the class. In addition, on the class of all frames, the basic algorithm is conservative: on purely modal input it computes interpolants in which the hybrid syntactic machinery does not occur.

## 1 Introduction

Craig's interpolation lemma (if  $\varphi \rightarrow \psi$  is valid, then  $\varphi \rightarrow \theta$  and  $\theta \rightarrow \psi$  are valid, for  $\theta$  a formula constructed using only primitive symbols which occur both in  $\varphi$  and  $\psi$ ) fails for many propositional and first order modal logics. For elementary modal logics, interpolants exist in the first order correspondence language. (An elementary modal logic is the set of modal formulas valid on an elementary class of frames.) This suggests that the modal language is not rich enough for the structures it is interpreted in. Hybrid logic expands modal logic in a natural and elementary way without making it equivalent to the first order correspondence language. So we can ask whether interpolants which do not exist in the modal language can already be expressed in the hybrid language. This paper provides a positive answer. We show that interpolants can be computed from a tableau proof for formulas (in both propositional and first order hybrid languages) interpreted on any elementary class  $K$  of frames satisfying the following property

- (1)                       $\mathfrak{F}$  is in  $K$  if and only if all point generated subframes of  $\mathfrak{F}$  are in  $K$ .

---

\*We thank Clemens Grabmayer for his careful reading of the manuscript and his valuable suggestions for improvements. Research supported by NWO grants 612-062-001 and 612.000.106. This paper was written during a visit to the Langue et Dialogue team, LORIA, Nancy, France. The visit was funded by INRIA, Lorraine.

Hence for any orthodox modal logic of such a class of frames, interpolants are guaranteed to exist in the hybrid language.

A desirable property of an algorithm computing interpolants is that if a modal interpolant exists, the algorithm computes that instead of a hybrid formula. Propositional modal logic  $\mathbf{K}$  has the interpolation property. In this case, the algorithm can be adapted to produce modal formulas as interpolants.

That general interpolation results are possible in hybrid logic is known: the propositional case is dealt with in [1], the first order in [2]. But this earlier work is model-theoretic. The point of the present paper is to show that hybrid logic permits general interpolation results to be proved constructively.

We proceed as follows. Section 2 is a short introduction to the hybrid logic  $\mathcal{HL}(@, \downarrow)$ . The algorithm for computing interpolants (for the logic of the class of all frames) is given in section 3. Section 4 establishes a hybrid logic analogue of the Goldblatt–Thomason theorem and extends the algorithm to hybrid logics of frame classes satisfying property (1). In section 5 we add a garbage collection mechanism to our basic algorithm which ensures that on purely modal input it computes purely modal interpolants. First order modal logic is the topic of section 6. In the concluding section we return to the topic of matching syntax and semantics and the importance of being able to name states in the object language.

This paper has been influenced by the work of Melvin Fitting. First, Fitting [9] repairs interpolation in first order  $\mathbf{S5}$  by moving to modal languages in which it is possible to quantify over propositional symbols, an approach which bears a superficial resemblance to the use of hybridization. Actually, as is discussed in [2], there are important differences between the methods, and the results proved by means of hybridization are more general than Fitting’s. Nonetheless, Fitting also shows that his approach yields interpolation constructively. Our earlier model-theoretic work ignored this aspect of interpolation, and the present paper was written to fill the gap. Second, the method we use to prove interpolation draws heavily on Fitting’s approach to first order interpolation as presented in [8]. This is a modification of the symmetric Gentzen method used by Smullyan [13].

## 2 Hybrid logic

In this section we define the syntax and semantics of the hybrid logic  $\mathcal{H}(\downarrow, @)$  and note some of its basic properties. Our introduction is necessarily brief: for further background see Chapter 7 of [3], or the Hybrid Logic Homepage ([www.hylo.net](http://www.hylo.net)).

**Definition 2.1 (Language)** *Let  $\text{PROP} = \{p_1, p_2, \dots\}$  be a countable set of propositional variables,  $\text{NOM} = \{s_1, s_2, \dots\}$  a countable set of nominals, and  $\text{SVAR} = \{x_1, x_2, \dots\}$  a countable set of state variables (or world variables). We assume that  $\text{PROP}$ ,  $\text{NOM}$  and  $\text{SVAR}$  are pairwise disjoint. We call  $\text{SSYM} = \text{NOM} \cup \text{SVAR}$  the set of state symbols (or world symbols), and  $\text{ATOM} = \text{PROP} \cup \text{NOM} \cup \text{SVAR}$  the set of atoms. The well-formed formulas of the hybrid language (over the signature  $\langle \text{PROP}, \text{NOM}, \text{SVAR} \rangle$ ) are*

$$\varphi := \top \mid \perp \mid a \mid \neg\varphi \mid \varphi \wedge \varphi' \mid \varphi \vee \varphi' \mid \varphi \rightarrow \varphi' \mid \Box\varphi \mid \Diamond\varphi \mid \downarrow x_j.\varphi \mid @_l\varphi$$

where  $a \in \text{ATOM}$ ,  $x_j \in \text{SVAR}$  and  $l \in \text{SSYM}$ .

In what follows we assume that a signature  $\langle \text{PROP}, \text{NOM}, \text{SVAR} \rangle$ , and hence  $\mathcal{L}$ , has been fixed. We usually write  $p, q$  and  $r$  for propositional variables,  $n, s, t$  for nominals, and  $w, x, y$  and  $z$  for state variables.

Note that all three types of atomic symbols are *formulas*. Further, note that the above syntax is simply that of ordinary unimodal propositional modal logic extended by the clauses for  $\downarrow x_j.\varphi$  and  $@_l\varphi$ . Finally, the difference between nominals and state variables is simply this: nominals cannot be bound by  $\downarrow$ , whereas state variables can.

**Definition 2.2** *The notions of free and bound state variable are defined in the manner familiar from first order logic, with  $\downarrow$  as the only binding operator. Similarly, other syntactic notions (such as substitution, and of a state symbol  $t$  being substitutable for  $x$  in  $\varphi$ ) are defined in the same way as the corresponding notions in first order logic. We use  $\varphi[l'/l]$  to denote the formula obtained by replacing all free instances of the state symbol  $l$  by the state symbol  $l'$ , such that no unwanted bindings occur.*

*A sentence is a formula containing no free state variables. Furthermore, a formula is pure if it contains no propositional variables, and nominal-free if it contains no nominals.*

**Definition 2.3 (Semantics)** *A (hybrid) model  $\mathfrak{M}$  for  $\mathcal{L}$  is a triple  $\mathfrak{M} = \langle M, R, V \rangle$  such that  $M$  is a non-empty set,  $R$  a binary relation on  $M$ , and  $V : \text{PROP} \cup \text{NOM} \rightarrow \text{Pow}(M)$  such that for all nominals  $s \in \text{NOM}$ ,  $V(s)$  is a singleton subset of  $M$ . We call the elements of  $M$  states (though the reader can think of them as worlds or times, if this terminology is more familiar),  $R$  the accessibility relation, and  $V$  the valuation. A frame is a pair  $\mathfrak{F} = \langle M, R \rangle$ ; that is, a frame is a model without a valuation.*

*An assignment  $g$  for  $\mathfrak{M}$  is a mapping  $g : \text{SVAR} \rightarrow M$ . Given an assignment  $g$ , we define  $g_m^x$  (an  $x$ -variant of  $g$ ) by  $g_m^x(x) = m$  and  $g_m^x(y) = g(y)$  for  $x \neq y$ .*

*Let  $\mathfrak{M} = \langle M, R, V \rangle$  be a model,  $m \in M$ , and  $g$  an assignment. For any atom  $a$ , let  $[V, g](a) = \{g(a)\}$  if  $a$  is a state variable, and  $V(a)$  otherwise. Then the forcing relation is defined as follows*

$$\begin{array}{ll}
\mathfrak{M}, g, m \Vdash \top & \\
\mathfrak{M}, g, m \Vdash a & \text{iff } m \in [V, g](a), a \in \text{ATOM} \\
\mathfrak{M}, g, m \Vdash \neg\varphi & \text{iff } \mathfrak{M}, g, m \not\Vdash \varphi \\
\mathfrak{M}, g, m \Vdash \varphi \wedge \psi & \text{iff } \mathfrak{M}, g, m \Vdash \varphi \text{ and } \mathfrak{M}, g, m \Vdash \psi \\
\mathfrak{M}, g, m \Vdash \Box\varphi & \text{iff } \forall m' (Rmm' \Rightarrow \mathfrak{M}, g, m' \Vdash \varphi) \\
\mathfrak{M}, g, m \Vdash \downarrow x.\varphi & \text{iff } \mathfrak{M}, g_m^x, m \Vdash \varphi \\
\mathfrak{M}, g, m \Vdash @_l\varphi & \text{iff } \mathfrak{M}, g, m' \Vdash \varphi, \text{ where } [V, g](l) = \{m'\}, l \in \text{SSYM}.
\end{array}$$

*The meaning of the formulas  $\perp, \varphi \vee \varphi', \varphi \rightarrow \varphi, \Diamond\varphi$  can be derived by their usual definitions in terms of  $\top, \neg, \wedge$  and  $\Box$  (where  $\Diamond\varphi = \neg\Box\neg\varphi$ ).*

*We write  $\mathfrak{M} \models \varphi$  iff for all  $m$ , for all  $g$ ,  $\mathfrak{M}, g, m \Vdash \varphi$ . A formula  $\varphi$  is satisfiable if there is a model  $\mathfrak{M}$ , an assignment  $g$  on  $\mathfrak{M}$ , and a state  $m \in M$  such that  $\mathfrak{M}, g, m \Vdash \varphi$ . A formula  $\varphi$  is valid if for all models  $\mathfrak{M}$ ,  $\mathfrak{M} \models \varphi$ .*

The first five clauses are essentially the standard Kripke forcing relation for propositional modal logic; in fact the only difference is that whereas the standard definition relativizes semantic evaluation to states  $m$ , we relativize to variable assignments  $g$  as well. Note that the clause for atoms covers all three types of symbol (propositional variables, nominals, and state variables) and that given any model  $\mathfrak{M}$  and assignment  $g$ , any state symbol (whether it is a nominal or a state variable) will be forced at a *unique* state; this is an immediate consequence of the way we defined valuations and assignments. Thus nominals and state variables act as names for states: they “name” the unique state they are true at. The  $\downarrow$

binds state variables to the state where evaluation is being performed (the *current state*), and  $@_l$  shifts evaluation to the state named by  $l$  (to put it more simply:  $@_l\varphi$  asserts that  $\varphi$  holds at the state labelled  $l$ ). Just as in first order logic, if  $\varphi$  is a *sentence* it is irrelevant which assignment  $g$  is used to perform evaluation:  $\mathfrak{M}, g, m \Vdash \varphi$  for *some* assignment  $g$  iff  $\mathfrak{M}, g, m \Vdash \varphi$  for *all* assignments  $g$ . Hence for sentences the relativization to assignments of the forcing relation can be dropped, and we simply write  $\mathfrak{M}, m \Vdash \varphi$ .

**Hybrid logic  $\neq$  first order logic** The expressive power of  $\mathcal{H}(\downarrow, @)$  is characterised semantically and syntactically in [1] by an extension of the well known standard translation of modal logic into first order logic. These translations show that both modal logic and hybrid logic can be viewed as fragments of first order logic. An interpolation theorem stated for a fragment of first order logic is only meaningful if it is a proper fragment (otherwise the theorem would hold immediately by Craig’s result for first order logic). The semantic characterization of hybrid logic shows that the hybrid language is a proper fragment: for instance for first order formulas like  $\exists xP(x)$ ,  $\exists xR(x, x)$  or  $\forall xR(yx)$  there is no equivalent hybrid formula.

**The tableau calculus** A complete hybrid tableau system (see [4]) for  $\mathcal{H}(\downarrow, @)$  over the class of all frames is presented in Table 1. The tableau proof system works in the expected way. To fix terminology we recall that a tableau branch is *closed* if it contains  $\perp$  or  $\varphi$  and  $\neg\varphi$ , where  $\varphi$  is a formula. A tableau is closed if each branch is closed. *Atoms in a tableau* are formulas of the form  $@_sp$ ,  $@_st$  and  $@_s\Diamond t$ , for  $p$  a propositional variable and  $s, t$  nominals. A branch is *atomically closed* if it closes on an atom and its negation. We note that the hybrid system is complete even when branches may only be atomically closed. Since it will make some proofs easier, we assume atomic closure throughout the paper. A *(tableau) proof* of a hybrid sentence  $\varphi$  is a closed tableau beginning with  $\neg@_s\varphi$ , where  $s$  is a nominal not occurring in  $\varphi$ . Inspection of the tableau rules shows that any tableau started with a sentence contains no free state variables.

### 3 Interpolation calculation rules

**Theorem 3.1** (1) For all  $\mathcal{HL}(@, \downarrow)$  sentences  $\varphi$  and  $\psi$ , the following are equivalent,

(i)  $\varphi \rightarrow \psi$  is valid.

(ii) there exists a  $\mathcal{HL}(@, \downarrow)$  sentence  $\theta$  whose propositional symbols and nominals all occur in both  $\varphi$  and  $\psi$  such that  $\varphi \rightarrow \theta$  and  $\theta \rightarrow \psi$  are valid.

(2) Given a closed hybrid tableau for  $\varphi \rightarrow \psi$ , the interpolant  $\theta$  can be computed effectively.

The proof method is the one used by Fitting [8] for first order logic, and the following terminology and definitions are taken from this source.

Here’s the basic idea. A tableau proof of  $\varphi \rightarrow \psi$  starts with  $\neg@_s(\varphi \rightarrow \psi)$ , and then  $@_s\varphi$  and  $\neg@_s\psi$  are added. After this, any formula added to the tableau must be a descendant of either  $@_s\varphi$  or  $\neg@_s\psi$ . So we enhance the tableau machinery to keep track of the ancestor of each formula. Think of  $\varphi$  as “left” and  $\psi$  as “right”, which are their respective positions in  $\varphi \rightarrow \psi$ . We symbolise this by writing  $L(@_s\varphi)$  and  $R(\neg@_s\psi)$  and systematically use this notation throughout (these labels are just bookkeeping devices and play no other role). Call expressions of the form  $L(\varphi)$  and  $R(\varphi)$  *biased formulas*.

Constant rules		
$\frac{\neg @_s \perp}{\perp}$	$\frac{\neg @_s \top}{\top}$	
Negation rules		
$\frac{\neg @_s \neg \varphi}{@_s \varphi}$	$\frac{@_s \neg \varphi}{\neg @_s \varphi}$	
Conjunctive rules		
$\frac{@_s(\varphi \wedge \psi)}{@_s \varphi \quad @_s \psi}$	$\frac{\neg @_s(\varphi \vee \psi)}{\neg @_s \varphi \quad \neg @_s \psi}$	$\frac{\neg @_s(\varphi \rightarrow \psi)}{@_s \varphi \quad \neg @_s \psi}$
Disjunctive rules		
$\frac{@_s(\varphi \vee \psi)}{@_s \varphi \mid @_s \psi}$	$\frac{\neg @_s(\varphi \wedge \psi)}{\neg @_s \varphi \mid \neg @_s \psi}$	$\frac{@_s(\varphi \rightarrow \psi)}{\neg @_s \varphi \mid @_s \psi}$
Diamond rules		
$\frac{@_s \diamond \varphi}{@_s \diamond t \quad @_t \varphi}$	$\frac{\neg @_s \Box \varphi}{@_s \diamond t \quad \neg @_t \varphi}$	$t$ is new to the branch,
Box rules		
$\frac{@_s \Box \varphi \quad @_s \diamond t}{@_t \varphi}$	$\frac{\neg @_s \diamond \varphi \quad @_s \diamond t}{\neg @_t \varphi}$	
@ rules		
$\frac{@_s @_t \varphi}{@_t \varphi}$	$\frac{\neg @_s @_t \varphi}{\neg @_t \varphi}$	$\frac{[s \text{ on the branch}]}{@_s s}$ [Ref]
		$\frac{@_s t \quad @_s \varphi}{@_t \varphi}$ [Nom]
		$\frac{@_s t \quad @_r \diamond s}{@_r \diamond t}$ [Bridge]
Downarrow rules		
$\frac{@_s \downarrow x. \varphi}{@_s \varphi [s/x]}$	$\frac{\neg @_s \downarrow x. \varphi}{\neg @_s \varphi [s/x]}$	

Table 1: Hybrid tableau system

We can extend the tableau rules from Table 1 to biased sentences in the obvious way. For instance, the rule for disjunctions yields the following two biased rules:

$$\frac{L(@_s(\varphi \vee \psi))}{L(@_s\varphi) \mid L(@_s\psi)} \quad \frac{R(@_s(\varphi \vee \psi))}{R(@_s\varphi) \mid R(@_s\psi)}$$

The other rules with one premise are treated analogously. The reflexivity rule has to be changed slightly to record where the introduced nominal came from:

$$\frac{L(\varphi)}{L(@_s s)} \quad \frac{R(\varphi)}{R(@_s s)} \quad \text{with } s \text{ occurring in } \varphi$$

Now for the two premise rules. These all display the same behaviour: the major premiss determines the label of the conclusion. To save space, let  $X$  and  $Y$  be variables ranging over  $\{L, R\}$ . Then the box rules can be summarised as:

$$\frac{Y(@_s \diamond t), X(@_s \Box \varphi)}{X(@_t \varphi)} \quad \frac{Y(@_s \diamond t), X(\neg @_s \diamond \varphi)}{X(\neg @_t \varphi)}$$

The Nom and Bridge rules become:

$$\frac{Y(@_s t), X(@_s \varphi)}{X(@_t \varphi)} \text{ [Nom]} \quad \frac{Y(@_s t), X(@_r \diamond s)}{X(@_r \diamond t)} \text{ [Bridge]}$$

Call a tableau constructed with these rules a *biased tableau*. A branch of such a tableau is closed if it contains a contradiction, ignoring the  $L$  and  $R$  symbols. Obviously any normal tableau can be converted into a biased tableau and vice-versa.

The interpolant for the validity  $\varphi \rightarrow \psi$  will be extracted from the closed biased tableau for  $\{L(@_s \varphi), R(\neg @_s \psi)\}$ . We need a final bit of terminology before we can show how this is done. We say that a formula  $I$  is an interpolant for the finite set  $\{L(\varphi_1), \dots, L(\varphi_n), R(\psi_1), \dots, R(\psi_k)\}$ , provided  $I$  is an interpolant in the sense of Theorem 3.1 for the formula  $(\varphi_1 \wedge \dots \wedge \varphi_n) \rightarrow (\neg \psi_1 \vee \dots \vee \neg \psi_k)$ . (Take the empty conjunction to be  $\top$  and the empty disjunction to be  $\perp$ .)

**Notation.**  $S \xrightarrow{int} I$  means that  $I$  is an interpolant for the set  $S$  of biased formulas.

By this definition and the fact that  $s$  is new to  $\varphi$  and  $\psi$ ,  $\downarrow w. I[w/s]$  is an interpolant for  $\varphi \rightarrow \psi$  whenever  $I$  is an interpolant for the set  $\{L(@_s \varphi), R(\neg @_s \psi)\}$ .

We are ready to give Fitting-style interpolation calculation rules. We start with those for branches  $S$  that are closed:

$$\begin{array}{ll} S \cup \{L(\varphi), L(\neg \varphi)\} & \xrightarrow{int} \perp \\ S \cup \{R(\varphi), R(\neg \varphi)\} & \xrightarrow{int} \top \\ S \cup \{L(\varphi), R(\neg \varphi)\} & \xrightarrow{int} \varphi \\ S \cup \{R(\varphi), L(\neg \varphi)\} & \xrightarrow{int} \neg \varphi \\ S \cup \{L(\perp)\} & \xrightarrow{int} \perp \\ S \cup \{R(\perp)\} & \xrightarrow{int} \top \end{array}$$

Next the propositional cases which are the same for  $R$  and  $L$ . We indicate the label with an  $X$ . We only list one conjunctive case, the others are treated in the same way.

$$\begin{array}{c}
\frac{S \cup \{X(\top)\} \xrightarrow{int} I}{S \cup \{X(\neg @_s \perp)\} \xrightarrow{int} I} \quad \frac{S \cup \{X(\perp)\} \xrightarrow{int} I}{S \cup \{X(\neg @_s \top)\} \xrightarrow{int} I} \\
\frac{S \cup \{X(@_s \varphi)\} \xrightarrow{int} I}{S \cup \{X(\neg @_s \neg \varphi)\} \xrightarrow{int} I} \quad \frac{S \cup \{X(\neg @_s \varphi)\} \xrightarrow{int} I}{S \cup \{X(@_s \neg \varphi)\} \xrightarrow{int} I} \\
\frac{S \cup \{X(@_s \varphi), X(@_s \psi)\} \xrightarrow{int} I}{S \cup \{X(@_s(\varphi \wedge \psi))\} \xrightarrow{int} I}
\end{array}$$

The disjunctive rules differ depending on the sign. Again we list only one case.

$$\begin{array}{c}
\frac{S \cup \{L(@_s \varphi)\} \xrightarrow{int} I_1 \quad S \cup \{L(@_s \psi)\} \xrightarrow{int} I_2}{S \cup \{L(@_s(\varphi \vee \psi))\} \xrightarrow{int} I_1 \vee I_2} \\
\frac{S \cup \{R(@_s \varphi)\} \xrightarrow{int} I_1 \quad S \cup \{R(@_s \psi)\} \xrightarrow{int} I_2}{S \cup \{R(@_s(\varphi \vee \psi))\} \xrightarrow{int} I_1 \wedge I_2}
\end{array}$$

The correctness of these propositional rules is shown in Fitting [8].

Now the rules for  $\downarrow$ , the first two  $@$  rules and for the modalities:

$$\begin{array}{c}
\frac{S \cup \{X(@_s \varphi[s/w])\} \xrightarrow{int} I}{S \cup \{X(@_s \downarrow w. \varphi)\} \xrightarrow{int} I} \\
\frac{S \cup \{X(@_t \varphi)\} \xrightarrow{int} I}{S \cup \{X(@_s @_t \varphi)\} \xrightarrow{int} I} \\
\frac{S \cup \{X(@_s \diamond t), X(@_t \varphi)\} \xrightarrow{int} I}{S \cup \{X(@_s \diamond \varphi)\} \xrightarrow{int} I} \quad t \text{ does not occur in } S \\
\frac{S \cup \{X(@_s \diamond t), X(@_t \varphi)\} \xrightarrow{int} I}{S \cup \{X(@_s \diamond t), X(@_s \square \varphi)\} \xrightarrow{int} I}
\end{array}$$

To sum up: for  $\downarrow, \diamond$  and the first two  $@$  rules the label does not matter. The rule for box is also the same for  $R$  and  $L$  if both the box formula and the formula  $@_s \diamond t$  have the same label. Note that we have not displayed the rules for  $\neg @_s \downarrow w. \varphi, \neg @_s @_t \varphi, \neg @_s \diamond \varphi$  and  $\neg @_s \square \varphi$ , but these are, of course, analogous to the rules for their positive counterparts. We leave the reader to verify the correctness of these rules and turn to the most demanding cases<sup>1</sup>.

These cases concern the  $@$ -rules Ref, Nom and Bridge and the Box rule when the box formula and the “firing” formula  $@_s \diamond t$  have *different* labels. We start with the latter. For

<sup>1</sup>In many ways these are analogous to some of the rules for quantifiers in the classical setting. The analogy is best seen when considering  $@_s \square \varphi$  as the relativized universal statement  $\forall y(Rsy \rightarrow \varphi(y))$ . An application of the Box rule corresponds in a first order tableau proof to a universal instantiation, an application of a disjunctive rule, and an immediate closure of one of the branches.

every label there are two cases. We assume that the variable  $w$  does not occur in  $I$ . For easy reference we have given the rules names:

$$\begin{array}{l}
\text{(Box R)} \quad \frac{S \cup \{L(@_s \diamond t), R(@_t \varphi)\} \xrightarrow{\text{int}} I}{S \cup \{L(@_s \diamond t), R(@_s \square \varphi)\} \xrightarrow{\text{int}} \begin{cases} @_s \diamond t \wedge I & \text{if } t \text{ occurs in } \psi_1, \dots, \psi_k \text{ or } \varphi \\ @_s \diamond \downarrow w. I[w/t] & \text{otherwise} \end{cases}} \\
\text{(Box L)} \quad \frac{S \cup \{R(@_s \diamond t), L(@_t \varphi)\} \xrightarrow{\text{int}} I}{S \cup \{R(@_s \diamond t), L(@_s \square \varphi)\} \xrightarrow{\text{int}} \begin{cases} @_s \diamond t \rightarrow I & \text{if } t \text{ occurs in } \varphi_1, \dots, \varphi_n \text{ or } \varphi \\ @_s \square \downarrow w. I[w/t] & \text{otherwise} \end{cases}}
\end{array}$$

We verify the correctness of the Box L rule. Suppose  $S \cup \{R(@_s \diamond t), L(@_t \varphi)\} \xrightarrow{\text{int}} I$ . Then both  $\varphi_1 \wedge \dots \wedge \varphi_n \wedge @_t \varphi \rightarrow I$  and  $\psi_1 \wedge \dots \wedge \psi_k \wedge @_s \diamond t \rightarrow \neg I$  are valid, and all symbols of  $I$  are common to  $\{\varphi_1, \dots, \varphi_n, @_t \varphi\}$  and  $\{\psi_1, \dots, \psi_k, @_s \diamond t\}$ .

First assume that  $t$  occurs in  $\varphi_1, \dots, \varphi_n$  or  $\varphi$ . The assumption  $\varphi_1 \wedge \dots \wedge \varphi_n \wedge @_t \varphi \rightarrow I$  and the validity  $@_s \square \varphi \wedge @_s \diamond t \rightarrow @_t \varphi$  yields the validity of  $\varphi_1 \wedge \dots \wedge \varphi_n \wedge @_s \square \varphi \rightarrow (@_s \diamond t \rightarrow I)$ . On the other side, the assumption  $\psi_1 \wedge \dots \wedge \psi_k \wedge @_s \diamond t \rightarrow \neg I$  immediately yields  $\psi_1 \wedge \dots \wedge \psi_k \wedge @_s \diamond t \rightarrow @_s \diamond t \wedge \neg I$ , whose consequent equals  $\neg(@_s \diamond t \rightarrow I)$ . Also all symbols in  $@_s \diamond t \rightarrow I$  occur in both  $\{\varphi_1 \dots \varphi_n, @_s \square \varphi\}$  and  $\{\psi_1 \dots \psi_k, @_s \diamond t\}$ , since the only symbol which may cause problems,  $t$ , was assumed to be among the  $\varphi_i$  and  $\varphi$ .

Next, assume that  $t$  does not occur in  $\varphi_1, \dots, \varphi_n$  or  $\varphi$ . Because  $t$  is not among the  $\varphi_i$  and  $\varphi$  and  $w$  is new to the formula,  $\varphi_1 \wedge \dots \wedge \varphi_n \wedge @_t \varphi \rightarrow I$  is equivalent to  $\varphi_1 \wedge \dots \wedge \varphi_n \wedge @_w \varphi \rightarrow I[w/t]$ . Applying necessitation for  $@_s, \square$  and  $\downarrow w$ , this yields the validity of

$$@_s \square \downarrow w. (\varphi_1 \wedge \dots \wedge \varphi_n \wedge @_w \varphi \rightarrow I[w/t]).$$

Because  $w$  is new, and all formulas in the antecedent are  $@$ -formulas, this distributes to

$$\varphi_1 \wedge \dots \wedge \varphi_n \wedge @_s \square \downarrow w. @_w \varphi \rightarrow @_s \square \downarrow w. I[w/t].$$

Because  $w$  does not occur in  $\varphi$ ,  $@_s \square \downarrow w. @_w \varphi$  is equivalent to  $@_s \square \varphi$ . Hence  $\varphi_1 \wedge \dots \wedge \varphi_n \wedge @_s \square \varphi \rightarrow @_s \square \downarrow w. I[w/t]$ , as desired.

Applying necessitation for  $@_t$  to  $\psi_1 \wedge \dots \wedge \psi_k \wedge @_s \diamond t \rightarrow \neg I$ , and distributing (using the fact that all formulas in the antecedent are  $@$ -formulas) yields  $\psi_1 \wedge \dots \wedge \psi_k \wedge @_s \diamond t \rightarrow \neg @_t I$ . Together with the validity  $@_s \diamond t \rightarrow (@_s \square \downarrow w. I[w/t] \rightarrow @_t I)$ , this yields  $\psi_1 \wedge \dots \wedge \psi_k \wedge @_s \diamond t \rightarrow \neg @_s \square \downarrow w. I[w/t]$ , as desired. As  $t$  was the only possible cause of problems in the interpolant, and that is bound away, the new formula is still an interpolant. Verification of the Box R rule is symmetrical and left to the reader.

Finally we come to the rules for handling equalities: these are displayed in Table 2. The Reflexivity rule is straightforward. The rules Nom and Bridge display the same case distinction as the Box rules. Again when both premises have the same label, the rules for  $L$  and  $R$  are the same. We leave the verification of these rules to the reader; the proofs are similar in spirit (though somewhat simpler) to the one given above for the Box rule.

We have given all the rules, and their correctness is clear, hence Theorem 3.1 has been proved.



(Refl)	$\frac{S \cup \{X(\varphi), X(@_s s)\} \xrightarrow{int} I}{S \cup \{X(\varphi)\} \xrightarrow{int} I}$
(Nom)	$\frac{S \cup \{X(@_s t), X(@_t \varphi)\} \xrightarrow{int} I}{S \cup \{X(@_s t), X(@_s \varphi)\} \xrightarrow{int} I}$
(Bridge)	$\frac{S \cup \{X(@_s t), X(@_r \diamond t)\} \xrightarrow{int} I}{S \cup \{X(@_s t), X(@_r \diamond s)\} \xrightarrow{int} I}$
(Nom R)	$\frac{S \cup \{L(@_s t), R(@_t \varphi)\} \xrightarrow{int} I}{S \cup \{L(@_s t), R(@_s \varphi)\} \xrightarrow{int} \begin{cases} @_s t \wedge I & \text{if } t \text{ occurs in } \psi_1, \dots, \psi_k, \varphi \\ I[s/t] & \text{otherwise} \end{cases}}$
(Nom L)	$\frac{S \cup \{R(@_s t), L(@_t \varphi)\} \xrightarrow{int} I}{S \cup \{R(@_s t), L(@_s \varphi)\} \xrightarrow{int} \begin{cases} @_s t \rightarrow I & \text{if } t \text{ occurs in } \varphi_1, \dots, \varphi_n, \varphi \\ I[s/t] & \text{otherwise} \end{cases}}$
(Bridge R)	$\frac{S \cup \{L(@_s t), R(@_r \diamond t)\} \xrightarrow{int} I}{S \cup \{L(@_s t), R(@_r \diamond s)\} \xrightarrow{int} \begin{cases} @_s t \wedge I & \text{if } t \text{ occurs in } \psi_1, \dots, \psi_k \\ I[s/t] & \text{otherwise} \end{cases}}$
(Bridge L)	$\frac{S \cup \{R(@_s t), L(@_r \diamond t)\} \xrightarrow{int} I}{S \cup \{R(@_s t), L(@_r \diamond s)\} \xrightarrow{int} \begin{cases} @_s t \rightarrow I & \text{if } t \text{ occurs in } \varphi_1, \dots, \varphi_n \\ I[s/t] & \text{otherwise} \end{cases}}$

Table 2: Calculation rules for Reflexivity, Nom and Bridge.

**Example** The web page <http://www.science.uva.nl/~marx/hybrid/interpolation.ps> contains the computation of the interpolant for the validity

$$(2) \quad (\diamond p \wedge \diamond \neg p) \rightarrow (\Box(q \rightarrow n) \rightarrow \diamond \neg q).$$

Here  $n$  is a nominal and  $p, q$  are propositional variables. The formula says that if a state has two successors, then if it has at most one  $q$  successor, it has at least one  $\neg q$  successor. In [1] this formula is used as a counterexample to interpolation for the hybrid language without the binder  $\downarrow$ . (In fact a very similar counterexample shows failure of interpolation in the logic of the difference operator. In this logic nominals are definable [5].) Interpolants for (2) are

$$(3) \quad \downarrow w. \diamond \downarrow u. @_w \diamond \downarrow v. (\neg @_u v),$$

and the more economical  $\downarrow w. \diamond \downarrow y. @_w \diamond \neg y$ . Both express the property of having two successors. The computed interpolant for  $\{L(@_s(\diamond p \wedge \diamond \neg p)), R(\neg @_s(\Box(q \rightarrow n) \rightarrow \diamond \neg q))\}$  is  $@_s \diamond \downarrow u. (@_s \diamond u \wedge \top \wedge @_s \diamond \downarrow v. (@_s \diamond v \wedge \neg @_u v))$ . This is equivalent to  $@_s \diamond \downarrow u. @_s \diamond \downarrow v. (\neg @_u v)$  (see also Section 5). Now binding away the nominal  $s$  yields (3).

**Lyndon interpolation** We can strengthen the interpolation theorem for hybrid logic to a Lyndon interpolation theorem, which relates positive and negative occurrences of propositional symbols in the interpolant to their occurrences in  $L(\varphi)$  and  $R(\neg\psi)$  (for a precise formulation see Fitting [8]). The proof is straightforward: show that the calculation rules preserve Lyndon interpolants. For the propositional rules this is as in [8]. For all other rules it suffices to check that they don't change the sign of the occurrences of propositional symbols. Note that Lyndon interpolation only holds for propositional symbols, not for nominals. In

$$[@_s \Box(p \rightarrow \neg q) \wedge @_s \diamond(t \wedge p)] \rightarrow \neg @_t q,$$

the nominal  $t$  will occur in any interpolant but it occurs positively on the left and negatively on the right.

**Multimodal logic** The interpolation calculus can also be extended to multimodal logic: add diamond and box rules for every modal operator  $\langle R \rangle$ . In this case, we can also interpolate on the modalities: only common modalities may occur in the interpolant. Call such an interpolant a multimodal interpolant. An inspection of the rules shows that all rules preserve multimodal interpolants: only the closing rules and the box rules can introduce modal operators but only if they occur both left and right. Whence Theorem 3.1 also holds in this stronger form for multimodal hybrid logic.

**Theorem 3.2** *For multimodal hybrid logic, the interpolation theorem holds in the stronger sense that interpolants may only contain modalities which occur both in the antecedent and the consequent of the premise.*

## 4 Hybrid logics of elementary frame classes

Theorem 3.1 only gives a method of constructing hybrid interpolants over the class of all frames. However, as we shall now show, it is straightforward to adapt the machinery introduced in the previous section to compute interpolants for the hybrid logic of many modally interesting frame classes.

**Theorem 4.1** *Let  $\mathbf{K}$  be an elementary class of frames of the form  $(W, R)$  satisfying for all frames  $\mathfrak{F}, \mathfrak{F}$  in  $\mathbf{K}$  if and only if all point generated subframes of  $\mathfrak{F}$  are in  $\mathbf{K}$ . Then the following are equivalent for all hybrid sentences  $\varphi, \psi$ :*

- (i)  $\varphi \rightarrow \psi$  is valid on the class  $\mathbf{K}$ .
- (ii) there exists a hybrid sentence  $\theta$  whose propositional symbols and nominals all occur in both  $\varphi$  and  $\psi$  such that  $\varphi \rightarrow \theta$  and  $\theta \rightarrow \psi$  are valid on the class  $\mathbf{K}$ .

In the proof of this theorem we use the following hybrid analogue of the Goldblatt-Thomason theorem:

**Theorem 4.2** *Let  $\mathbf{K}$  be an elementary class of frames of the form  $(W, R)$ . The following are equivalent:*

- (i) For all frames  $\mathfrak{F}, \mathfrak{F}$  is in  $\mathbf{K}$  if and only if all point generated subframes of  $\mathfrak{F}$  are in  $\mathbf{K}$ .
- (ii)  $\mathbf{K}$  is definable by a set of nominal free pure hybrid sentences.

PROOF. From (ii) to (i) follows because hybrid formulas are invariant under generated submodels ([1] Thm 3.11). For the other direction, assume  $\mathbf{K}$  has property (i). Let  $\Theta(\mathbf{K})$  be the set of nominal free pure hybrid sentences valid in  $\mathbf{K}$ . Let  $\mathfrak{F} \models \Theta(\mathbf{K})$ . We will show that  $\mathfrak{F}$  is in  $\mathbf{K}$ . For  $w$  in the domain of  $\mathfrak{F}$ , let  $\mathfrak{F}_w$  denote the subframe of  $\mathfrak{F}$  generated from  $w$ . Then  $\mathfrak{F}_w \models \Theta(\mathbf{K})$  by [1] Thm 3.11. We show that for all  $w$ ,  $\mathfrak{F}_w$  is in  $\mathbf{K}$ ; this yields the result because  $\mathbf{K}$  has property (i). Let  $\Delta_w$  be the set of nominal free pure hybrid sentences  $\varphi$  such that  $\mathfrak{F}_w, w \Vdash \varphi$ . (That is,  $\Delta_w$  is the set of nominal free pure hybrid sentences valid on  $\mathfrak{F}_w$  at  $w$ ; as such formulas contain neither non-logical symbols nor free variables the choice of valuation and assignment is irrelevant.) Note that  $\Theta(\mathbf{K}) \subseteq \Delta_w$ . Hence each  $\varphi \in \Delta_w$  is satisfiable on a  $\mathbf{K}$  frame (for if not, then  $\neg\varphi \in \Theta(\mathbf{K})$ , whence  $\mathfrak{F}_w, w \Vdash \varphi$  and  $\mathfrak{F}_w, w \Vdash \neg\varphi$ ). Then by compactness  $\Delta_w$  is satisfiable on a  $\mathbf{K}$  frame  $\mathfrak{G}$  at some  $v$ . As before,  $\mathfrak{G}_v, v \Vdash \Delta_w$  for  $\mathfrak{G}_v$  generated from  $v$ , hence because  $\mathbf{K}$  has property (i),  $\mathfrak{G}_v$  is also in  $\mathbf{K}$ . Let  $\mathfrak{F}_w^+$  and  $\mathfrak{G}_v^+$  be  $\omega$  saturated elementary extensions of  $\mathfrak{F}_w$  and  $\mathfrak{G}_v$ , respectively. Without loss of generality we may assume that  $\mathfrak{F}_w^+$  and  $\mathfrak{G}_v^+$  are generated from  $w$  and  $v$ , respectively. By elementary equivalence,  $\mathfrak{G}_v^+$  is in  $\mathbf{K}$ . We use the following result to finish the argument (this can be shown as in the proof of [1] Thm 3.7.):

Let  $\mathfrak{M}_m^+$  and  $\mathfrak{N}_n^+$  be point generated  $\omega$  saturated models. The following are equivalent:

- (i)  $\mathfrak{M}_m^+, m$  and  $\mathfrak{N}_n^+, n$  validate the same nominal free pure hybrid sentences.
- (ii)  $\mathfrak{M}_m^+, m$  and  $\mathfrak{N}_n^+, n$  validate the same first order formulas in the signature  $\{R\}$  in at most one free variable.

$\mathfrak{F}_w^+, w$  and  $\mathfrak{G}_v^+, v$  validate the same nominal free pure hybrid sentences by construction. Thus  $\mathfrak{F}_w^+$  and  $\mathfrak{G}_v^+$  validate the same first order sentences in the signature  $\{R\}$ . Because  $\mathbf{K}$  is an elementary class of frames in the signature  $\{R\}$ , then  $\mathfrak{F}_w^+$  is in  $\mathbf{K}$ , and so, by elementary embedding,  $\mathfrak{F}_w$  is in  $\mathbf{K}$ . QED

PROOF OF THEOREM 4.1. Let  $\mathbf{K}$  be as in the hypothesis of the theorem. Let  $\Gamma$  be the set of pure nominal free hybrid sentences which defines the class  $\mathbf{K}$  ( $\Gamma$  exists by Theorem 4.2). Then adding the following rules to the hybrid tableaux calculus yields completeness (see [4]) of the hybrid logic of the class  $\mathbf{K}$ :

$$\frac{}{\textcircled{a}_s \gamma} \quad \text{for } \gamma \in \Gamma \text{ and } s \text{ on the branch.}$$

Such rules are easy to handle in the calculus which computes interpolants. The conclusion gets its label from the label of a formula in which  $s$  occurred:

$$\frac{X(\varphi) \text{ with } s \text{ in } \varphi}{X(@_s\gamma)}.$$

The extra rule for computing the interpolants for implications valid on the class  $\mathbf{K}$  now becomes

$$\frac{S \cup \{X(\varphi), X(@_s\gamma)\} \xrightarrow{int} I}{S \cup \{X(\varphi)\} \xrightarrow{int} I}.$$

Verification of this rule is trivial because  $\gamma$  itself does not contain propositional symbols or nominals. QED

## 5 Constructing interpolants for modal input

As the example in section 3 should have made clear, on input formulas containing nominals, interpolants containing downarrows and variables are inevitable. But what does the algorithm produce when given a *purely modal* formula as input? A formula  $\varphi$  is purely modal if it contains no nominals and no state variables — so we are asking: what happens when we input an orthodox modal formula to our hybrid-logical algorithm? This is an interesting question: the (orthodox) modal logic  $\mathbf{K}$  (the logic of the class of all frames) enjoys interpolation, thus purely modal interpolants exist. A conservative algorithm should produce such interpolants in this case.

We shall now add a “cleaning up” step to our basic algorithm which deletes unnecessary  $@$  symbols and downarrows and produces purely modal interpolants for purely modal input. We view this as a sign that the basic algorithm is well designed, and that the intermediate use of the extra hybrid logical language inside proofs (for purely modal input) is essentially a handy device that can (over the class of all frames) be eliminated from the end result. More precisely, we will show:

**Theorem 5.1** *One can add a cleaning-up rule to the basic hybrid interpolation calculus which makes it produce purely modal interpolants when given a purely modal formula as input.*

*On purely multimodal input the interpolant will in addition only contain shared modalities.*

The following two lemmas are used in the proof of the theorem. Both are proved by induction on the length of the derivation. These lemmas are easier to formulate assuming interpolants of the form  $@_s\Diamond t \rightarrow I$  are read as  $\neg @_s\Diamond t \vee I$ , and this assumption is made throughout. We will from now on call nominals parameters to emphasize the fact that they only occur inside proofs.

**Lemma 5.2** *If a purely modal formula is valid, it has an atomically closed tableau proof in which:*

- (i) *none of the  $@$ - or  $\downarrow$ -rules is used;*
- (ii) *parameters only occur in formula position in formulas of the form  $@_n\Diamond m$ ;*
- (iii) *there is never closure on  $@_n\Diamond m$  atoms.*

**Lemma 5.3** *During the interpolant construction, the interpolant  $I$  is always in modal normal form: that is, constructed using  $\wedge, \vee, @_s \diamond \downarrow w$  and  $@_s \square \downarrow w$  from atoms  $@_s p, @_s t$  and  $@_s \diamond t$  and their negations, and  $\top$  and  $\perp$ . Here  $s, t$  are nominals or variables,  $w$  a variable and  $p$  a propositional letter.*

The next lemma is an improvement of Lemma 5.3 for purely modal input.

**Lemma 5.4** *Let  $T$  be an atomically closed tableau without applications of  $@$ - and  $\downarrow$ -rules starting with  $L(@_c \varphi)$  and  $R(\neg @_c \psi)$ , for  $\varphi, \psi$  purely modal in which the rewriting described in Figure 1 is performed after application of the Box rules. Then during the interpolant construction  $I$  is always in the following modal normal form: constructed using  $\wedge, \vee$  from  $p, @_n p, @_n \diamond m$  and their negations, and  $\top, \perp$ , and  $@_n \diamond \theta, @_n \square \theta$ , with  $p$  a proposition letter,  $n, m$  parameters, and  $\theta$  purely modal.*

PROOF OF LEMMA 5.4. By induction on the length of the derivation. By Lemma 5.2.(iii) the closure rules can only produce  $\top, \perp, @_p$  and  $\neg @_n p$  in  $I$ . All other rules except the Box R and Box L rules when they introduce a downarrow and a variable produce an interpolant in the required format (with the assumption that  $@_s \diamond t \rightarrow I$  is read as  $\neg @_s \diamond t \vee I$ ). Consider an application of Box L. Thus assume that  $I$  is of the required format,  $S \cup \{R(@_s \diamond t), L(@_t \varphi)\} \xrightarrow{int} I$  holds and  $t$  does not occur in  $\varphi_1, \dots, \varphi_n$  or  $\varphi$ . Then, by the Box L rule,  $@_s \square \downarrow w. I[w/t]$  is an interpolant for  $S \cup \{R(@_s \diamond t), L(@_s \square \varphi)\}$ .

**Claim.** If  $@_n \diamond m$  occurs in  $I[w/t]$ , and  $w$  occurs in  $@_n \diamond m$ , then  $n = s$  and  $m = w$ .

With this claim, the rewrite rules turn  $@_s \square \downarrow w. I[w/t]$  into an equivalent interpolant  $I$  which is in the format required by the lemma.

We now prove the claim. First assume  $@_n \diamond w$  occurs in  $I[w/t]$ . Then  $@_n \diamond t$  occurs in  $I$  and it occurred in the tableau. Also, by assumption,  $@_s \diamond t$  occurs in the tableau. But then  $n = s$ , because every  $@_n \diamond m$  formula is introduced in the tableau using a new parameter  $m$  and none of the  $@$ -rules have been applied.

Next assume that  $@_w \diamond m$  occurs in  $I[w/t]$ . Then  $@_t \diamond m$  occurred in the tableau. As  $m$  occurs in the interpolant,  $m$  must occur in left and right formulas in  $S$ . The first formula in a tableau in which a parameter occurs is the one in which it is introduced, so that is  $@_t \diamond m$ . Thus  $@_t \diamond m$  is in  $S$ . Because we assumed  $t$  does not occur in the  $\varphi_1, \dots, \varphi_n$  it must be  $R(@_t \diamond m)$ . Inspection of the tableau rules shows that the only way of getting  $m$  at the left is with a formula  $L(@_t \square \varphi)$  and application of the Box L rule. But then  $L(@_t \square \varphi)$  must also still be in  $S$ . A contradiction with the assumption that  $t$  not in the  $\varphi_1, \dots, \varphi_n$ .

This proves the claim and with that the case of Box L. The argument for an application of Box R is symmetrical. QED

Now Theorem 5.1 follows by a simple argument.

PROOF OF THEOREM 5.1. Suppose  $\models \varphi \rightarrow \psi$ , with  $\varphi, \psi$  purely modal. By the previous Lemma, the tableau started with  $L(@_s \varphi), R(\neg @_s \psi)$  produces an interpolant which is a boolean combination of formulas  $@_s \theta$ , with  $\theta$  purely modal. A distribution of  $@_s$  outside the boolean operators then yields an interpolant  $@_s \theta$  with  $\theta$  purely modal. As  $s$  does not occur in  $\varphi, \psi, \theta$ , the validity of  $@_s \varphi \rightarrow @_s \theta$  and  $@_s \theta \rightarrow @_s \psi$  yields the validity of  $\varphi \rightarrow \theta$  and  $\theta \rightarrow \psi$ .

The result on multimodal input follows from the present result and Theorem 3.2. QED

After an application of a Box rule in which the new interpolant is either  $@_s \diamond \downarrow w. I[w/t]$  or  $@_s \square \downarrow w. I[w/t]$ , immediately perform the following rewriting of the interpolant into an equivalent formula:

**Step 1** Distribute  $@_s \diamond \downarrow w$  and  $@_s \square \downarrow w$  over disjunctions and conjunctions using the validities

$$\begin{aligned} @_s \diamond \downarrow w. (\varphi \vee \psi) &\equiv @_s \diamond \downarrow w. \varphi \vee @_s \diamond \downarrow w. \psi \\ @_s \square \downarrow w. (\varphi \wedge \psi) &\equiv @_s \square \downarrow w. \varphi \wedge @_s \square \downarrow w. \psi. \end{aligned}$$

**Step 2** Distribute downarrows over conjunction and disjunction using the validities

$$\downarrow w. (\varphi \vee \psi) \equiv \downarrow w. \varphi \vee \downarrow w. \psi \text{ and } \downarrow w. (\varphi \wedge \psi) \equiv \downarrow w. \varphi \wedge \downarrow w. \psi.$$

**Step 3** Remove vacuous  $\downarrow w$  repeatedly using the validities

$$\begin{aligned} \downarrow w. @_w \varphi &\equiv \varphi \text{ whenever } w \text{ does not occur in } \varphi \\ \downarrow w. \varphi &\equiv \varphi \text{ whenever } w \text{ does not occur in } \varphi \\ @_s \diamond (\downarrow w. @_s \diamond w \wedge \varphi) &\equiv @_s \diamond \varphi \\ @_s \diamond (\downarrow w. \neg @_s \diamond w \wedge \varphi) &\equiv \perp \\ @_s \square (\downarrow w. \neg @_s \diamond w \vee \varphi) &\equiv @_s \square \varphi \\ @_s \square (\downarrow w. @_s \diamond w \vee \varphi) &\equiv \top. \end{aligned}$$

**Step 4** Pull out @ formulas from under the scope of the modalities repeatedly using the validities

$$\begin{aligned} @_s \diamond @_t \varphi &\equiv @_t \varphi \wedge @_s \diamond \top \\ @_s \diamond (@_t \varphi \wedge \psi) &\equiv @_t \varphi \wedge @_s \diamond \psi \\ @_s \square @_t \varphi &\equiv @_t \varphi \vee @_s \square \perp \\ @_s \square (@_t \varphi \vee \psi) &\equiv @_t \varphi \vee @_s \square \psi. \end{aligned}$$

The rules in step 3 and 4 should be read modulo commutativity and associativity of  $\vee$  and  $\wedge$ . Given this and the assumption that  $I$  is of the form stated in the lemma, the rewrite system is guaranteed to terminate.

Figure 1: Rewrite system for purely modal input

## 6 First order modal logic

The interpolation property fails in many first order modal logics. For instance, Fine [6] shows how interpolation fails in every constant domain first order modal logic between **K** and **S5**. In [2] we gave interpolants in the first order hybrid language for Fine's counterexamples and proved a general interpolation result model-theoretically. Here we shall show that interpolants for first order hybrid logic can be obtained constructively.

The language of quantified hybrid logic we shall work with was introduced in [4]. It is obtained by adding first order atomic formulas and the quantifiers  $\exists x$  and  $\forall x$  to  $\mathcal{H}(\downarrow, @)$ . The terms in the language are first order variables, constants interpreted as rigid designators, and unary functions which take a nominal or world variable as argument. The latter are used to model non rigid designators. Atomic formulas are created as usual from these terms using predicate symbols of any arity and the equality symbol. This defines the quantified hybrid language **QHL**.

Meaning is defined by a straightforward extension of the definition for propositional hybrid logic. We shall give a constant domain semantics. A **QHL** model is a structure  $(M, R, D, I_{nom}, I_m)_{m \in M}$  such that

- $(M, R)$  is a modal frame;
- $I_{nom}$  is a function assigning members of  $M$  to nominals;
- for every  $m \in M$ ,  $(D, I_m)$  is an ordinary first order model such that
  - $I_m(c) = I_{m'}(c)$ , for all  $m, m' \in M$  and constants  $c$ ;
  - $I_m(q) \in D$ , for  $q$  a unary function symbol;
  - $I_m(P) \subseteq {}^k D$ , for  $P$  a  $k$ -ary predicate symbol.

To interpret formulas with free variables we extend the assignments  $g$  to assign elements of  $D$  to variables ranging over individuals. Given a model and an assignment  $g$ , the interpretation of terms  $t$ , denoted by  $\bar{t}$ , is defined as

$$\begin{array}{lll}
 \bar{x} & = & g(x) & \text{for } x \text{ a variable} \\
 \bar{c} & = & I_m(c) & \text{for } c \text{ a constant and some } m \in M \\
 \overline{q(n)} & = & \begin{cases} I_{I_{nom}(n)}(q) & \text{if } n \text{ a nominal} \\ I_{g(n)}(q) & \text{if } n \text{ a state variable.} \end{cases} & \text{for } q \text{ a unary function symbol}
 \end{array}$$

Formulas are now interpreted as usual (we only list the new clauses):

$$\begin{array}{ll}
 \mathfrak{M}, g, m \Vdash P(t_1, \dots, t_n) & \iff \langle \bar{t}_1, \dots, \bar{t}_n \rangle \in I_m(P) \\
 \mathfrak{M}, g, m \Vdash t_i = t_j & \iff \bar{t}_i = \bar{t}_j \\
 \mathfrak{M}, g, m \Vdash \exists x \varphi & \iff \mathfrak{M}, g_d^x, m \Vdash \varphi, \text{ for some } d \in D \\
 \mathfrak{M}, g, m \Vdash \forall x \varphi & \iff \mathfrak{M}, g_d^x, m \Vdash \varphi, \text{ for all } d \in D.
 \end{array}$$

The tableau calculus for **QHL** consists of the rules for hybrid propositional logic (Table 1) plus the extra rules for the quantifiers and equality in Table 3. Completeness is established in [4]. The rules for the quantifiers and the Ref and RR rules come from Fitting [8]. The DD rule states that the non-rigid designator  $q$  denotes the same object in states  $s$  and  $t$  whenever

<b>Existential rules</b>				
$\frac{\@_s \exists x \varphi(x)}{\@_s \varphi(c)}$	$\frac{\neg \@_s \forall x \varphi(x)}{\neg \@_s \varphi(c)}$			
<b>Universal rules</b>				
$\frac{\@_s \forall x \varphi(x)}{\@_s \varphi(t)}$	$\frac{\neg \@_s \exists x \varphi(x)}{\neg \@_s \varphi(t)}$			
<b>QHL Equality rules</b>				
$\frac{}{t = t} [\text{Ref}]$	$\frac{t = u, \varphi(t)}{\varphi[u]} [\text{RR}],$	$\frac{\@_s t}{q(s) = q(t)} [\text{DD}]$	$\frac{\@_s (t_i = t_j)}{t_i = t_j} [\text{@=}]$	$\frac{\neg \@_s (t_i = t_j)}{\neg (t_i = t_j)} [\text{@=}]$
<p>In the existential rules, <math>c</math> is a first order parameter which is new to the branch.          In the universal and Ref rules, <math>t</math> is a term occurring on the branch. In RR, <math>\varphi[u]</math> indicates that some (not necessarily all) occurrences of <math>t</math> in <math>\varphi</math> are replaced by <math>u</math>. In the DD rule, <math>q(s)</math> must be on the branch.</p>				

Table 3: Extra tableau rules for **QHL**

$s$  and  $t$  denote the same state. This rule has as a side condition that  $q(s)$  should be on the branch. The @= rules specify that equality is a rigid predicate (its meaning is independent of the state of evaluation).

**Theorem 6.1** (1) For all **QHL** sentences  $\varphi$  and  $\psi$ , the following are equivalent:

- (i)  $\varphi \rightarrow \psi$  is valid.
  - (ii) there exists an **QHL** sentence  $\theta$  whose relational symbols, constants and nominals all occur in both  $\varphi$  and  $\psi$  such that  $\varphi \rightarrow \theta$  and  $\theta \rightarrow \psi$  are valid.
- (2) Given a closed hybrid tableau for  $\varphi \rightarrow \psi$ , the interpolant  $\theta$  can be computed effectively.

The theorem follows by extending the interpolation algorithm to the new rules. The new rules yield labels as one would expect, though care should be taken with the reflexivity rule and the rule DD. Reflexivity is handled analogously the hybrid reflexivity rule (that is, we need to record where the introduced term came from). As for DD, the result gets its label from the label of the formula in which the definite description occurs:

$$\frac{X(@_s t), Y(\varphi)}{Y(q(s) = q(t))} \quad \text{with } q(s) \text{ occurring in } \varphi.$$

Now for the rules for computing interpolants. The quantifier rules are those given by Fitting [8], and we do not repeat them here. The reflexivity and the replacement rule work precisely the same way as the hybrid reflexivity and Nom rules, respectively. The @= rules are straightforward:

$$\frac{S \cup \{X(@_s(t_i = t_j)), X(t_i = t_j)\} \xrightarrow{\text{int}} I}{S \cup \{X(@_s(t_i = t_j))\} \xrightarrow{\text{int}} I} \quad \frac{S \cup \{X(\neg @_s(t_i = t_j)), X(\neg t_i = t_j)\} \xrightarrow{\text{int}} I}{S \cup \{X(\neg @_s(t_i = t_j))\} \xrightarrow{\text{int}} I}$$



The DD rule displays the same case distinction as the Nom rule:

$$\frac{S \cup \{X(@_s t), X(\varphi), X(q(s) = q(t))\} \xrightarrow{int} I}{S \cup \{X(@_s t), X(\varphi)\} \xrightarrow{int} I}$$

$$\frac{S \cup \{R(@_s t), L(\varphi), L(q(s) = q(t))\} \xrightarrow{int} I}{S \cup \{R(@_s t), L(\varphi)\} \xrightarrow{int} \begin{cases} @_s t \rightarrow I & \text{if } t \text{ occurs in } \varphi_1, \dots, \varphi_n, \varphi \\ I[s/t] & \text{otherwise} \end{cases}}$$

$$\frac{S \cup \{L(@_s t), R(\varphi), R(q(s) = q(t))\} \xrightarrow{int} I}{S \cup \{L(@_s t), R(\varphi)\} \xrightarrow{int} \begin{cases} @_s t \wedge I & \text{if } t \text{ occurs in } \psi_1, \dots, \psi_k, \varphi \\ I[s/t] & \text{otherwise} \end{cases}}$$

For the verification of the quantifier rules we refer to [8]. Verification of the other rules is either trivial or follows a similar argument as for the Box rule given before. (Note that the side-condition on DD is needed for the verification.) Thus Theorem 6.1 is proven.

**More results** The strengthening to Lyndon interpolation is immediate because the added rules keep the signs of the relation symbols. In the multimodal case, interpolation holds as in Theorem 3.2. This follows at once, because the new rules do not introduce modalities. Also, just as in the propositional case (Theorem 4.1) interpolation holds for every frame class which is definable with pure nominal free hybrid sentences. Here we assumed constant domains, but the results go through for varying, contracting or expanding domains as well [2].

## 7 Conclusion

Orthodox modal logic is badly behaved with respect to interpolation. In the simple propositional logics typically assumed in philosophical discussion (for example **K**, **T**, **S4**, and **S5**) interpolation goes through smoothly (see for example Fitting [7] or Rautenberg [12]). But even in the propositional case, once one leaves these simple logics, interpolation results are hard to come by (it fails, for example, in the modal logic of Church-Rosser frames, and see Maksimova [10] for results above **S4**). And in the first order case, positive interpolation results are few and far between (Fine [6] showed that none of the first order constant domain modal logics between **K** and **S5** has interpolation).

Failure of interpolation (and in particular of the closely associated Beth definability property) is often taken to be a sign of a mismatch between syntax and semantics. The basic idea driving hybrid logic is that there is an *obvious* mismatch between orthodox modal syntax and semantics: states (or worlds) are central to modal semantics, but they are nowhere represented in orthodox modal syntax. Hybrid logicians argue that the mechanisms for state reference that they introduce (nominals, @, and ↓) fix these problems in a way that is both non-trivial and modally natural. It is non-trivial because their addition does not result in a notational variant of first order logic. It is modally natural because (as the semantic characterisation of  $\mathcal{HL}(@, \downarrow)$  shows) the internal perspective on models typical of modal logic is preserved. We view the constructive interpolation results presented here as firm technical support for the claim that hybrid logic adds precisely the syntactic machinery that is missing from modal logic.

To conclude, an open problem. The following result is proved model-theoretically in [11]: Every canonical propositional modal logic which is complete with respect to a first order universal Horn definable class of frames enjoys the interpolation property. We would like to improve on this in two ways: give a constructive proof using the present calculus and the garbage collection technique used for modal input and replace the requirement of canonicity by requiring that the class of frames enjoys property (1)

## References

- [1] C. Areces, P. Blackburn, and M. Marx. Hybrid logics: Characterization, interpolation and complexity. *Journal of Symbolic Logic*, 66(3):977–1010, 2001.
- [2] C. Areces, P. Blackburn, and M. Marx. Repairing the interpolation lemma in quantified modal logic. Technical Report PP–2001–19, Institute for Logic, Language and Computation, University of Amsterdam, 2001. To appear in *Annals of Pure and Applied Logic*.
- [3] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2001.
- [4] P. Blackburn and M. Marx. Tableaux for quantified hybrid logic. In U. Egly and C. Fermüller, editors, *Automated Reasoning with Analytic Tableaux and Related Methods, TABLEAUX 2002*, number 2381 in LNAI, pages 38–52. Springer Verlag, 2002.
- [5] M. de Rijke. The modal logic of inequality. *Journal of Symbolic Logic*, 57:566–584, 1992.
- [6] K. Fine. Failures of the interpolation lemma in quantified modal logic. *Journal of Symbolic Logic*, 44(2):201–206, 1979.
- [7] M. Fitting. *Proof Methods for Modal and Intuitionistic Logics*. Number 169 in Synthese Library. Reidel, Dordrecht, 1983.
- [8] M. Fitting. *First Order Logic and Automated Theorem Proving (second edition)*. Springer Verlag, 1996.
- [9] M. Fitting. Interpolation for first order S5. *Journal of Symbolic Logic*, 67:621–634, 2002.
- [10] L. Maksimova. Amalgamation and interpolation in normal modal logics. *Studia Logica*, L(3/4):457–471, 1991.
- [11] M. Marx and Y. Venema. *Multi-dimensional Modal Logic*. Applied Logic Series. Kluwer Academic Publishers, 1997.
- [12] W. Rautenberg. Modal tableau calculi and interpolation. *Journal of Philosophical Logic*, 12:403–423, 1983.
- [13] R. Smullyan. *First Order Logic*. Springer–Verlag, 1968.